SOLUTIONS TO SELECTED
EXERCISES IN

THE LOGIC BOOK

Fifth Edition

MERRIE BERGMANN *Smith College*
JAMES MOOR *Dartmouth College*
JACK NELSON *Arizona State University*

Higher Education

Boston Burr Ridge, IL Dubuque, IA New York
San Francisco St. Louis Bangkok Bogotá Caracas Kuala Lumpur
Lisbon London Madrid Mexico City Milan Montreal New Delhi
Santiago Seoul Singapore Sydney Taipei Toronto

The McGraw·Hill Companies

Mc Graw Hill Higher Education

Solutions to Selected Exercises in
THE LOGIC BOOK
Merrie Bergmann
James Moor
Jack Nelson

Published by McGraw-Hill, an imprint of The McGraw-Hill Companies, Inc., 1221 Avenue of the Americas, New York, NY 10020.

1 2 3 4 5 6 7 8 9 0 EVA/EVA 0 9 8 7 6 5 4 3

ISBN 0-07-334314-5

Vice president and Editor-in-chief: *Michael Ryan*
Publisher: *Beth Mejia*
Sponsoring editor: *Mark Georgiev*
Editorial assistant: *Briana Porco*
Marketing manager: *Pamela S. Cooper*
Production editor: *Leslie LaDow*
Media project manager: *Thomas Brierly*
Production supervisor: *Tandra Jorgensen*
Compositor: *Aptara®, Inc.*
Typeface: *10/12 New Baskerville*

www.mhhe.com

CONTENTS

SOLUTIONS TO CHAPTER 1 1

SOLUTIONS TO CHAPTER 2 10

SOLUTIONS TO CHAPTER 3 20

SOLUTIONS TO CHAPTER 4 40

SOLUTIONS TO CHAPTER 5 88

SOLUTIONS TO CHAPTER 6 134

SOLUTIONS TO CHAPTER 7 148

SOLUTIONS TO CHAPTER 8 162

SOLUTIONS TO CHAPTER 9 198

SOLUTIONS TO CHAPTER 10 266

SOLUTIONS TO CHAPTER 11 310

SOLUTIONS TO SELECTED EXERCISES

CHAPTER ONE

Section 1.3E

1.a. This sentence does have a truth-value and does fall within the scope of this text. It is false if by ‘second President of the United States’ we mean the second person to hold the office of President as established by the Constitution of the United States. However, it is true if we mean the second person to bear the title ‘President of the United States’, as the Articles of Confederation, which predate the Constitution, established a loose union of states whose first and only president, John Hanson, did bear the title ‘President of the United States.

c. This is a request or command, as such it is neither true nor false, and therefore does not fall within the scope of this text.

e. This sentence does have a truth-value (it is true), and does fall within the scope of this text.

g. This sentence does have a truth-value and does fall within the scope of this text. It is false, as Bill Clinton is the President who immediately preceded George W. Bush.

i. This sentence is neither true nor false, for if it were true, then sentence m would be true, and if m is true then what it says, that m is false, is also true. And no sentence can be both true and false. See the answer to exercise m below.

k. This sentence gives advice and is neither true nor false. Hence it does not fall within the scope of this text.

m. This appears to be a straightforward, unproblematic claim. But it is not. In fact, it embodies a well-known paradox. For if what the sentence says is true, then the sentence itself is, as is claimed, false. And if what the sentence says is false, then the sentence is not false and therefore is true. So the sentence is true if and only if it is false, an impossibility. This is an example of the paradox of self-reference. We exclude paradoxical sentences from the scope of this text.

2.a. When Mike, Sharon, Sandy, and Vicky are all out of the office no important decisions get made.

Mike is off skiing.

Sharon is in Spokane.

Vicky is in Olympia and Sandy is in Seattle.

No decisions will be made today.

c. This passage does not express any obvious argument. It is best construed as a series of related claims about the people in the office in question.

e. This passage does not express any obvious argument. It is best construed as a series of related claims about the contents of a set of drawers.

g. This passage does not express an obvious argument, though it might be claimed that the last sentence, 'So why are you unhappy' is rhetorical and has here the force of 'So you should be happy', yielding the following argument:

The weather is perfect; the view is wonderful; and we're on vacation.

You should be happy.

i. Wood boats are beautiful but they require too much maintenance.

Fiberglass boats require far less maintenance, but they tend to be more floating bathtubs than real sailing craft.

Steel boats are hard to find, and concrete boats never caught on.

So there's no boat that will please me.

k. Everyone from anywhere who's anyone knows Barrett.

All those who know Barrett respect her and like her.

Friedman is from Minneapolis and Barrett is from Duluth.

Friedman doesn't like anyone from Duluth.

Either Friedman is a nobody or Minneapolis is a nowhere.

m. Whatever is required by something that is good is itself a good.

Being cured of cancer is a good.

Being cured of cancer requires having cancer.

Having cancer is a good.

o. When there are more than two political parties, support tends to split among the parties with no one party receiving the support of a majority of voters.

No party can govern effectively without majority support.

When there is only one political party, dissenting views are neither presented nor contested.

When there are two or more viable parties, dissenting views are presented and contested.

Only the two party system is compatible both with effective governance and with the presenting and contesting of dissenting views.

Section 1.4E

1.a. False. Many valid arguments have one or more false premise. Here is an example with two false premises:

All Doberman pinschers are friendly creatures.

All friendly creatures are dogs.

All Doberman pinschers are dogs.

c. True. By definition, a sound argument is a valid argument with true premises.

e. False. A valid argument all of whose premises are true cannot have a false conclusion. But if a valid argument has at least one false premises, it may well have a false conclusion. Here is an example:

Reptiles are mammals.

If reptiles are mammals, then reptiles are warm blooded.

Reptiles are warm blooded.

g. False. An argument may have true premises and a true conclusion and not be valid. Here is an example:

Chicago is in Illinois.

Madrid is in Spain.

i. False. A sound argument is, by definition, a valid argument with true premises. And every valid argument with true premises has a true conclusion.

Section 1.5E

1.a. This passage is best construed as a deductive argument with some unexpressed or assumed premises. These premises include: Mike is skiing somewhere other than the office. No one can be in Spokane, or Olympia, or Seattle and in the office in question. With these premises added, the argument is deductively valid. Without them, it is deductively invalid.

c. As noted in the answers to exercises **1.3.2E**, the passage in question expresses no plausible argument. Construed as a deductive argument it is deductively invalid (no matter which claim is taken as the conclusion). Construed as an inductive argument it is inductively weak, again no matter which claim is taken as the conclusion.

e. Same answer as c. above.

g. This passage can be construed as an argument (see answers to **1.3.2.E**). So construed it is deductively invalid but inductively plausible.

i. This passage can be construed as a deductive argument with suppressed or assumed premises. The missing premises can be expressed as: 'All the boats there are either wood or fiberglass or steel or concrete', and 'No boat will please me if it requires too much maintenance, is a floating bathtub, is hard to find, or is of a type that never taught on.' Even with these premises added the argument is deductively invalid, as it does not follow from the claim that fiberglass boats ''tend to be floating bathtubs'' that every fiberglass is a floating bathtub.

k. This argument is best construed as a deductive argument, and is deductively valid. Since Barrett is from Duluth, and Friedman doesn't like anyone from Duluth, Friedman doesn't like Barrett. Hence, by the first premise, either the place Friedman is from (Minneapolis) is a nowhere, or Friedman isn't anyone, i.e., is a nobody.

m. This is a valid deductive argument. The conclusion is, of course, false. So we know that a least one of the premises is false. The best candidate for this position is ''Whatever is required by something that is good is itself a good''.

o. This passage is best construed as a deductive argument. From the first and second premises it follows that effective governance is not possible when there are more than two political parties. From the third and the fourth premises it follows that there must be at least two political parties for dissenting views to be presented and contested. Whether the argument is deductively valid depends on how we construe the claim ‘Only the two-party system is compatible both with effective governance and with the presenting and contesting of dissenting views.’ It is invalid if we take this claim to mean that the two-party system is compatible both with effective governance and with the presenting and contesting of dissenting views. The argument is valid if we take the claim in question to mean only that all systems other than the two-party systems are not so compatible.

Section 1.6E

1.a. {Kansas City is in Missouri, St. Paul is in Minnesota, San Francisco is in California}

c. There is no such set. If all the members of a set are true, then it is clearly possible for all those members to be true, and the set is therefore consistent.

2.a. All the members of this set are true (The Dodgers have not been in Brooklyn for almost half a century. Here, in the Northwest, good vegetables are hard to find. And today, the day this answer is written, is hotter than yesterday.) Since all the members are true, it is clearly possible for all the members to be true. Therefore, the set is consistent.

c. All three members of this set are true, so the set is consistent.

e. It is possible for all four members of this set to be true. Imagine yourself driving home on a Monday afternoon with a nearly empty gas tank.

g. The set is inconsistent. If no one who fails “Poetry for Scientists” is bright and Tom failed that course, it follows that Tom is not bright. So, for every member of the set to be true Tom would have to both be bright (as “Tom, Sue, and Robin are all bright’ alleges), and not be bright. This is not possible.

i. This set is inconsistent. If Kennedy was the best President we ever had, it cannot be that Eisenhower was a better President than Kennedy, and vice-versa. So not all the members of the set can be true.

k. This set is consistent. What is being claimed is that everyone who likes film classics likes *Casablanca,* not that everyone who likes *Casablanca* likes all film classics. So, it is possible for Sarah to like *Casablanca* without liking (all) film classics. Similarly, Sarah can like *Casablanca* without liking Humphrey Bogart.

3.a. ‘Que será, será’ is a logically true sentence (of Spanish). It means ‘Whatever will be, will be.’ This sentence, taken literally, is logically true. (Were it not, there would have to be something that will be and will not be, an impossibility.)

c. ‘Eisenhower preceded Kennedy as President’ is true and is logically indeterminate. It is true because of facts about the American political system and how the voters voted in 1956 and 1960, not because of any principles of logic.

4.a. Logically indeterminate. Passing the bar exam does not involve, as a matter of logic, having gone to law school. Lincoln passed the bar examination but never went to law school.

c. Logically false. An MD is a Doctor of Medicine, so every MD is a doctor.

e. Logically true. Whoever Robin is and whatever the class is, she either will, or will not, make it to the class by starting time.

g. Logically false. If Bob knows everyone in the class, and Robin is in the class, it follows that he knows Robin, so if the first part of this claim is true, the last part, which claims Bob doesn't know Robin, must be false.

i. Logically true. Since ocean fish are a kind of fish, it follows from 'Sarah likes all kinds of fish' that she likes ocean fish.

k. Logically indeterminate. This claim is almost certainly true, given the very large number of people there are, but it is not a logical truth. If all but a handful of people were killed, then one of the survivors might love everyone, including him or herself, and not be lacking in discrimination.

5.a. No one will win.
There will be no winner.

c. Not possible. If one sentence is logically true and the other is logically indeterminate, then it is possible for the second sentence to be false and the former true (the former is always true), and hence the sentences are not logically equivalent.

e. Any pair of logically true sentences will satisfy this condition, for example 'A square has four sides' and 'A mother has a child (living or dead)'. Neither sentence can be false, so it is impossible that one is true and the other false.

6.a. These sentences are not logically equivalent. It can, and does, happen that a person loves someone who does not return that love.

c. These sentences are not logically equivalent. What one claims to be the case is not always actually the case. Tom may want to impress his new boss, a gourmet cook, but refuse to indulge when presented with a plate of raw shark.

e. These sentences are not logically equivalent. If the first is true, then both Bill and Mary will fail to get into law school. The second sentence makes a weaker claim, that one or the other will not get into law school. It, unlike the first sentence, will be true if Mary gets into law school but Bill does not.

g. These sentences are not logically equivalent. If the first is true, then there are no non-Mariner fans at the rally, but it does not follow that all the Mariner fans are there. And if the second is true, it does not follow that no non-Mariner fans are present.

i. These sentences are not logically equivalent. There is often a difference between what is reported and what is the case. If a strike is imminent but no newscast so reports, the second of the sentences is true but the first false. So too, newcasts, even taken collectively, often get it wrong, as when all news outlets reported that Dewey won the presidential election in 1948 when in fact Truman won that election.

k. These sentences are not logically equivalent. If the first is true, then at least one of the two, Sarah and Anna, will not be elected, and perhaps neither will be elected. That is, this sentence will be true if neither is elected. But in that case the second sentence, which claims that one or the other will be elected, will be false.

m. These sentences are not logically equivalent. The first may well be true (each of us can probably name at least one person we dislike). Given the truth of the first sentence, the second sentence may still be false, for we may each dislike different persons, and there may be no one universally disliked person.

o. These sentences are not logically equivalent. It is plausible that each of us does like at least one person, but it does not follow that there is someone we all like.

Section 1.7E

1.a. True. If a member of a set of sentences is logically false, then that member cannot be true, and hence it cannot be that all the members are true. So the set is logically inconsistent.

c. True. Sentences that are logically equivalent cannot have different truth-values. So if all the premises of an argument are true, and one of those premises is equivalent to the conclusion, then the conclusion must also be true. Hence, that argument cannot have true premises and a false conclusion. It is, therefore, deductively valid.

e. True. ‘Whatever will be, will be’ is logically true. Therefore, any argument that has it as a conclusion cannot have a false conclusion, and, hence, cannot have true premises and a false conclusion. Any such argument is, therefore, deductively valid.

g. False. An argument all of whose premises are logically true is valid if and only if its conclusion is also logically true. If the conclusion of such an argument is not logically true, then it is possible for the premises all to be true (as logical truths they are always true) and the conclusion false.

2.a. No. Such a person obviously has at least one false belief, but her or his mistake is about the facts of geography and/or of the political organization of the United States.

c. Normally logic cannot tell us whether a sentence is true or false, for most of the sentences we normally deal with, truth is a matter of how things are with the world. And, to determine whether or not a valid argument is sound, we do need to determine whether the premises are true. However, in one case logic can tell us that an argument is sound. This is where the argument is valid and all the premises are logical truths.

e. If an argument has a logical falsehood as one of its premises, it is impossible for that premises to be true. If one premise cannot be true, then surely it cannot be that all the premises are true, and it cannot be that all the premises are true and the conclusion false. So the argument must be deductively valid.

g. If an argument has a logical truth for its conclusion, it is impossible for that conclusion to be false. And if the conclusion cannot be false, then it obviously cannot be that the premises are true and the conclusion false. Hence such an argument is deductively valid, no matter what its premises are. But it will be sound only if those premises are true. So some such arguments are sound (those with true premises) and some are unsound (those with at least one false premise).

i. Yes. If the set with a million sentences is consistent, then it is possible for all of those sentences to be true. Now consider a set each of whose members is equivalent to at least one member of that first set. Sentences that are equivalent have the same truth-value. Therefore, if all the million members of the first set are true, all the sentences of the second set, each of which is equivalent to a member of the first set, will also be true. Therefore, the second set is also consistent.

CHAPTER TWO

Section 2.1E

1.a. Both Bob jogs regularly and Carol jogs regularly.

$B \& C$

c. Either Bob jogs regularly or Carol jogs regularly.

$B \vee C$

e. It is not the case that either Bob jogs regularly or Carol jogs regularly.

$\sim (B \vee C)$

[or]

Both it is not the case that Bob jogs regularly and it is not the case that Carol jogs regularly.

$\sim B \& \sim C$

g. If it is not the case that Carol jogs regularly then it is not the case that Bob jogs regularly.

$\sim C \supset \sim B$

i. Both (either Bob jogs regularly or Albert jogs regularly) and it is not the case that (both Bob jogs regularly and Albert jogs regularly).

$(B \vee A) \& \sim (B \& A)$

k. Both it is not the case that (either Carol jogs regularly or Bob jogs regularly) and it is not the case that Albert jogs regularly.

$\sim (C \vee B) \& \sim A$

m. Either Albert jogs regularly or it is not the case that Albert jogs regularly.

$A \vee \sim A$

2.a. Albert jogs regularly and so does Bob.
c. Either Albert or Carol jogs regularly.
e. Neither Albert nor Carol jogs regularly.
g. Bob jogs regularly and so does either Albert or Carol.
i. Albert, Carol, and Bob jog regularly.
k. Either Bob or Carol jogs regularly, or neither of them jogs regularly.

3. c and k are true; and a, e, g, and i are false.

4. Paraphrases

a. It is not the case that all joggers are marathon runners.
c. It is not the case that some marathon runners are lazy.
e. It is not the case that somebody is perfect.

Symbolizations

a. Using ‘A’ for ‘All joggers are marathon runners’:

$\sim A$

c. Using ‘L’ for ‘Some marathon runners are lazy’:

$\sim L$

e. Using ‘P’ for ‘Somebody is perfect’:

$\sim P$

5.a. If Bob jogs regularly then it is not the case that Bob is lazy.

$B \supset \sim L$

c. Bob jogs regularly if and only if it is not the case that Bob is lazy.

$B \equiv \sim L$

e. Carol is a marathon runner if and only if Carol jogs regularly.

$M \equiv C$

g. If (both Carol jogs regularly and Bob jogs regularly) then Albert jogs regularly.

$(C \,\&\, B) \supset A$

i. If (either it is not the case that Carol jogs regularly or it is not the case that Bob jogs regularly) then it is not the case that Albert jogs regularly.

$(\sim C \vee \sim B) \supset \sim A$

k. If (both Albert is healthy and it is not the case that Bob is lazy) then (both Albert jogs regularly and Bob jogs regularly).

$(H \,\&\, \sim L) \supset (A \,\&\, B)$

m. If it is not the case that Carol is a marathon runner then [Carol jogs regularly if and only if (both Albert jogs regularly and Bob jogs regularly)].

$\sim M \supset [C \equiv (A \,\&\, B)]$

o. If [both (both Carol is a marathon runner and it is not the case that Bob is lazy) and Albert is healthy] then [both Albert jogs regularly and (both Bob jogs regularly and Carol jogs regularly)].

$[(M \,\&\, \sim L) \,\&\, H] \supset [A \,\&\, (B \,\&\, C)]$

q. If (if Carol jogs regularly then Albert jogs regularly) then (both Albert is healthy and Carol is a marathon runner).

$(C \supset A) \supset (H \ \& \ M)$

s. If [if (either Carol jogs regularly or Bob jogs regularly) then Albert jogs regularly)] then (both Albert is healthy and it is not the case that Bob is lazy).

$[(C \vee B) \supset A] \supset (H \ \& \sim L)$

6.a. Either Bob is lazy or he isn't.
c. Albert jogs regularly if and only if he is healthy.
e. Neither Bob nor Carol jogs regularly.
g. If either Albert or Carol does not jog regularly, then Bob does.
i. Carol jogs regularly only if Albert does but Bob doesn't.
k. Carol does and does not jog regularly.
m. If Bob is lazy, then he is; but Bob jogs regularly.
o. If Albert doesn't jog regularly, then Bob doesn't jog regularly only if Carol doesn't.
q. Albert doesn't jog regularly, and Bob jogs regularly if and only if he is not lazy.

7.a. Both both it is not the case that men are from Mars and it is not the case that women are from Mars and both it is not the case that men are from Venus and it is not the case that women are from Venus.

$(\sim M \ \& \sim W) \ \& \ (\sim V \ \& \sim S)$

c. It is not the case that both Butch Cassidy escaped and the Sundance Kid escaped.

$\sim (B \ \& \ S)$

e. Either both that lady was cut in half and that lady was torn asunder or it was a magic trick.

$(H \ \& \ A) \vee M$

g. Either the prisoner will receive a life sentence or the prisoner will receive the death penalty.

$L \vee D$

8.

P	**Q**	**$(P \vee Q) \ \& \sim (P \ \& \ Q)$**	**$P \equiv \sim Q$**
T	**T**	**F**	**F**
T	**F**	**T**	**T**
F	**T**	**T**	**T**
F	**F**	**F**	**F**

Section 2.2E

1.a. Either the French team will win at least one gold medal or either the German team will win at least one gold medal or the Danish team will win at least one gold medal.

$$F \vee (G \vee D)$$

c. Both (either the French team will win at least one gold medal or either the German team will win at least one gold medal or the Danish team will win at least one gold medal) and (either [it is not the case that either the French team will win at least one gold medal or the German team will win at least one gold medal] or [either (it is not the case that either the French team will win at least one gold medal or the Danish team will win at least one gold medal) or (it is not the case that either the German team will win at least one gold medal or the Danish team will win at least one gold medal)]).

$$[F \vee (G \vee D)] \ \& \ (\sim (F \vee G) \vee [\sim (F \vee D) \vee \sim (G \vee D)])$$

e. Either both the French team will win at least one gold medal and the German team will win at least one gold medal or either both the French team will win at least one gold medal and the Danish team will win at least one gold medal or both the German team will win at least one gold medal and the Danish team will win at least one gold medal.

$$(F \ \& \ G) \vee [(F \ \& \ D) \vee (G \ \& \ D)]$$

g. Either both both the French team will win at least one gold medal and the German team will win at least one gold medal and it is not the case that the Danish team will win at least one gold medal or either both both the French team will win at least one gold medal and the Danish team will win at least one gold medal and it is not the case that the German team will win at least one gold medal or both both the German team will win at least one gold medal and the Danish team will win at least one gold medal and it is not the case that the French team will win at least one gold medal.

$$[(F \ \& \ G) \ \& \sim D] \vee ([(F \ \& \ D) \ \& \sim G] \vee [(G \ \& \ D) \ \& \sim F])$$

2.a. None of them will win a gold medal.

c. None of them will win a gold medal.

e. At least one of them will win a gold medal.

g. The French team will win a gold medal and exactly one of the other two teams will win a gold medal.

3.a. If either the French team will win at least one gold medal or either the German team will win at least one gold medal or the Danish team will win at least one gold medal then both the French team will win at least one gold medal and both the German team will win at least one gold medal and the Danish team will win at least one gold medal.

$$[F \vee (G \vee D)] \supset [F \,\&\, (G \,\&\, D)]$$

c. If the star German runner is disqualified then if the German team will win at least one gold medal then it is not the case that either the French team will win at least one gold medal or the Danish team will win at least one gold medal.

$$S \supset [G \supset \sim (F \vee D)]$$

e. The Danish team will win at least one gold medal if and only if both the French team is plagued with injuries and the star German runner is disqualified.

$$D \equiv (P \,\&\, S)$$

g. If the French team is plagued with injuries then if the French team will win at least one gold medal then both it is not the case that either the Danish team will win at least one gold medal or the German team will win at least one gold medal and it rains during most of the competition.

$$P \supset (F \supset [\sim (D \vee G) \,\&\, R])$$

4.a. If the German star is disqualified then the German team will not win a gold medal, and the star is disqualified.

c. The German team won’t win a gold medal if and only if the Danish as well as the French will win one.

e. If a German team win guarantees a French team win and a French team win guarantees a Danish team win then a German team win guarantees a Danish team win.

g. Either at least one of the three wins a gold medal or else the French team is plagued with injuries or the star German runner is disqualified or it rains during most of the competition.

5.a. If it is not the case that the author of *Robert’s Rules of Order* was a politician, then either the author of *Robert’s Rules of Order* was an engineer or the author of *Robert’s Rules of Order* was a clergyman.

Both the author of *Robert’s Rules of Order* was motivated to write the book by an unruly church meeting and it is not the case that the author of *Robert’s Rules of Order* was a clergyman.

Both it is not the case that the author of *Robert’s Rules of Order* was a politician and the author of *Robert’s Rules of Order* could not persuade a publisher that the book would make money forcing him to publish the book himself.

The author of *Robert’s Rules of Order* was an engineer.

E: The author of *Robert's Rules of Order* was an engineer.
C: The author of *Robert's Rules of Order* was a clergyman.
P: The author of *Robert's Rules of Order* was a politician.
M: The author of *Robert's Rules of Order* was motivated to write the book by an unruly church meeting.
F: The author of *Robert's Rules of Order* could not persuade a publisher that the book would make money forcing him to publish the book himself.

$$\begin{array}{l} \sim P \supset (E \lor C) \\ M \;\&\; \sim C \\ \sim P \;\&\; F \\ \hline E \end{array}$$

c. Either either the maid committed the murder or the butler committed the murder or the cook committed the murder.

Both (if the cook committed the murder then a knife was the murder weapon) and (if a knife was the murder weapon then it is not the case that either the butler committed the murder or the maid committed the murder).

A knife was the murder weapon.

The cook committed the murder.

M: The maid committed the murder.
B: The butler committed the murder.
C: The cook committed the murder.
K: A knife was the murder weapon.

$$\begin{array}{l} (M \lor B) \lor C \\ (C \supset K) \;\&\; (K \supset \sim (B \lor M)) \\ K \\ \hline C \end{array}$$

e. If the candidate is perceived as conservative then both it is not the case that the candidate will win New York and both the candidate will win California and the candidate will win Texas.

Both if the candidate has an effective advertising campaign then the candidate is perceived as conservative and the candidate has an effective advertising campaign.

Either both the candidate will win California and the candidate will win New York or either (both the candidate will win California and the candidate will win Texas) or (both the candidate will win New York and the candidate will win Texas).

P: The candidate is perceived as conservative.
N: The candidate will win New York.
C: The candidate will win California.
T: The candidate will win Texas.
E: The candidate has an effective advertising campaign.

$$\begin{array}{l} P \supset [\sim N \,\&\, (C \,\&\, T)] \\ (E \supset P) \,\&\, E \\ \hline (C \,\&\, N) \vee [(C \,\&\, T) \vee (N \,\&\, T)] \end{array}$$

Section 2.3E

1. Since we do not know how these sentences are being used (e.g., as premises, conclusions, or as isolated claims) it is best to symbolize those that are non-truth-functional compounds as atomic sentences of *SL*.

a. ‘It is possible that’ does not have a truth-functional sense. Thus the sentence should be treated as a unit and abbreviated by one letter, for example, ‘E’. Here ‘E’ abbreviates not just ‘Every family on this continent owns a television set’ but the entire original sentence, ‘It is possible that every family on this continent owns a television set’.

c. ‘Necessarily’ has scope over the entire sentence. Abbreviate the entire sentence by one letter such as ‘N’.

e. This sentence can be paraphrased as a truth-functional compound:

Both it is not the case that Tamara will stop by and Tamara promised to phone early in the evening

which can be symbolized as ‘$\sim B \,\&\, E$’, where ‘B’ abbreviates ‘Tamara will stop by’ and ‘E’ abbreviates ‘Tamara promised to phone early in the evening’.

g. ‘John believes that’ is not a truth-functional connective. Abbreviate the sentence by one letter, for example ‘J’.

i. ‘Only after’ has no truth-functional sense. Therefore abbreviate the entire sentence as ‘D’.

2.a. The paraphrase is

If the maid committed the murder then the maid believed her life was in danger.

If the butler committed the murder then (both the murder was done silently and it is not the case that the body was mutilated).

Both the murder was done silently and it is not the case that the maid's life was in danger.

The butler committed the murder if and only if it is not the case that the maid committed the murder.

The maid committed the murder.

Notice that 'The maid believed her life was in danger' (first premise) and 'The maid's life was in danger' (third premise) make different claims and cannot be treated as the same sentence. Further, since the subjunctive conditional in the original argument is a premise, it can be weakened and paraphrased as a truth-functional compound. Using the abbreviations

M: The maid committed the murder.
D: The maid believed that her life was in danger.
B: The butler committed the murder.
S: The murder was done silently.
W: The body was mutilated.
L: The maid's life was in danger.

the symbolized argument is

$$
\begin{array}{l}
M \supset D \\
B \supset (S \,\&\, \sim W) \\
S \,\&\, \sim L \\
B \equiv \sim M \\
\hline
M
\end{array}
$$

c. The paraphrase is

If (both Charles Babbage had the theory of the modern computer and Charles Babbage had modern electronic parts) then the modern computer was developed before the beginning of the twentieth century.

Both Charles Babbage lived in the early nineteenth century and Charles Babbage had the theory of the modern computer.

Both it is not the case that Charles Babbage had modern electronic parts and Charles Babbage was forced to construct his computers out of mechanical gears and levers.

If Charles Babbage had had modern electronic parts available to him then the modern computer would have been developed before the beginning of the twentieth century.

In the original argument subjunctive conditionals occur in the first premise and the conclusion. Since it is correct to weaken the premises but not the conclusion, the first premise, but not the conclusion, is given a truth-functional paraphrase. The conclusion will be abbreviated as a single sentence. Using the abbreviations

T: Charles Babbage had the theory of the modern computer.
E: Charles Babbage had modern electronic parts.
C: The modern computer was developed before the beginning of the twentieth century.
L: Charles Babbage lived in the early nineteenth century.
F: Charles Babbage was forced to construct his computers out of mechanical gears and levers.
W: If Charles Babbage had had modern electronic parts available to him then the modern computer would have been developed before the beginning of the twentieth century.

the paraphrase can be symbolized as

$$\begin{array}{l} (T \ \& \ E) \supset C \\ L \ \& \ T \\ \sim E \ \& \ F \\ \hline W \end{array}$$

Section 2.4E

1.a. True

c. False. The chemical symbol names or designates the metal copper, not the word ‘copper’.

e. False. The substance copper is not its own name.

g. False. The name of copper is not a metal.

2.a. The only German word mentioned is ‘Deutschland’ which has eleven letters.

c. The phrase ‘the German name of Germany’ here refers to the word ‘Deutschland’, so ‘Deutschland’ is mentioned here.

e. The word ‘Deutschland’ occurs inside single quotation marks in Exercise 2.e, so it is there being mentioned, not used.

3.a. A sentence of *SL*.

c. A sentence of *SL*.

e. A sentence of *SL*.

g. A sentence of *SL*.

i. A sentence of *SL*.

4.a. The main connective is ‘&’. The immediate sentential components are ‘~ A’ and ‘H’. ‘~ A & H’ is a component of itself. Another sentential component is ‘A’. The atomic sentential components are ‘A’ and ‘H’.

c. The main connective is ‘∨’. The immediate sentential components are ‘~ (S & G)’ and ‘B’. The other sentential components are ‘~ (S & G) ∨ B’ itself, ‘(S & G)’, ‘S’, and ‘G’. The atomic components are ‘B’, ‘S’, and ‘G’.

e. The main connective is the first occurrence of ‘∨’. The immediate sentential components are ‘(C ≡ K)’ and ‘(~ H ∨ (M & N))’. Additional sentential components are the sentence itself, ‘~ H’, ‘(M & N)’, ‘C’, ‘K’, ‘H’, ‘M’, and ‘N’. The last five sentential components listed are atomic components.

5.a. No. The sentence is a conditional, but not a conditional whose antecedent is a negation.

c. Yes. Here **P** is the sentence ‘A’ and **Q** is the sentence ‘~ B’.

e. No. The sentence is a negation, not a conditional.

g. No. The sentence is a negation, not a conditional.

i. Yes. Here **P** is ‘A ∨ ~ B’ and **Q** is ‘~ (C & ~ D)’.

6.a. ‘H’ can occur neither immediately to the left of ‘~’ nor immediately to the right of ‘A’. As a unary connective, ‘~’ can immediately precede but not immediately follow sentences of *SL*. Both ‘H’ and ‘A’ are sentences of *SL*, and no sentence of *SL* can immediately precede another sentence of *SL*.

c. ‘(’ may not occur immediately to the right of ‘A’, as a sentence of *SL* can be followed only by a right parentheses or by a binary connective. But ‘(’ may occur immediately to the left of ‘~’, as in ‘(~ A & B)’.

e. ‘[’ may not occur immediately to the right of ‘A’ but may occur immediately to the left of ‘~’, as it functions exactly as does ‘(’.

CHAPTER THREE

Section 3.1E

1.a. $2^1 = 2$
c. $2^2 = 4$

2.a.

	↓					
E	~	~	(E	&	~	E)
T	**F**	**T**	**T**	**F**	**F**	**T**
F	**F**	**T**	**F**	**F**	**T**	**F**

c.

			↓					
A	J	A	≡	[J	≡	(A	≡	J)]
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**
F	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**
F	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**

e.

									↓			
A	H	J	[~	A	∨	(H	⊃	J)]	⊃	(A	∨	J)
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**
T	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**
F	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**
F	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**F**
F	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**
F	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**F**

g.

						↓					
A	B	~	(A	∨	B)	⊃	(~	A	∨	~	B)
T	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**
T	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**
F	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**

i.

			↓							
B	E	H	~	(E	&	[H	⊃	(B	&	E)])
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**
T	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**
T	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**
F	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**
F	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**
F	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**F**
F	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**

k.

									↓						
D	E	F	~	[D	&	(E	∨	F)]	≡	[~	D	&	(E	&	F)]
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**
T	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**
T	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**
F	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**
F	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**
F	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**
F	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**

m.

												↓			
A	H	J	(A	∨	(~	A	&	(H	⊃	J)))	⊃	(J	⊃	H)
T	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**
T	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**
F	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
F	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**T**
F	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**
F	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**

3.a.

			↓								
A	B	C	~	[~	A	∨	(~	C	∨	~	B)]
F	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**

c.

						↓			
A	B	C	(A	⊃	B)	∨	(B	⊃	C)
F	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**

e.

						↓			
A	B	C	(A	≡	B)	∨	(B	≡	C)
F	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**

g.

									↓			
A	B	C	~	[B	⊃	(A	∨	C)]	&	~	~	B
F	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**

i.

												↓			
A	B	C	~	[~	(A	≡	~	B)	≡	~	A]	≡	(B	∨	C)
F	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**

4.a.

				↓			
D	F	G	F	∨	(G	∨	D)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**F**	**T**	**T**
T	**F**	**T**	**F**	**T**	**T**	**T**	**T**
T	**F**	**F**	**F**	**T**	**F**	**T**	**T**
F	**T**	**T**	**T**	**T**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**F**	**F**	**F**
F	**F**	**T**	**F**	**T**	**T**	**T**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**

c.

								↓														
D	F	G	[F	∨	(G	∨	D)]	&	(~	(F	∨	G)	∨	[~	(F	∨	D)	∨	~	(G	∨	D)])
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
T	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
T	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
F	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**
F	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**

e.

						↓							
D	F	G	(F	&	G)	∨	[(F	&	D)	∨	(G	&	D)]
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
T	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**
T	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**
F	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**
F	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**
F	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**

g.

									↓													
D	F	G	[(F	&	G)	&	~	D]	∨	([(F	&	D)	&	~	G]	∨	[(G	&	D)	&	~	F])
T	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**
T	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**
T	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**
F	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**
F	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**
F	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**

5.a.

								↓					
D	F	G	[F	∨	(G	∨	D)]	⊃	[F	&	(G	&	D)]
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**
T	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**
T	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**
F	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**
F	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**
F	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**

c.

					↓						
D	F	G	S	S	⊃	[G	⊃	~	(F	∨	D)]
T	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**
T	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**
T	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
T	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**
T	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**
F	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**
F	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**
F	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**
F	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**
F	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**
F	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**

e.

				↓			
D	P	S	D	≡	(P	&	S)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**F**	**T**	**F**	**F**
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**
T	**F**	**F**	**T**	**F**	**F**	**F**	**F**
F	**T**	**T**	**F**	**F**	**T**	**T**	**T**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**
F	**F**	**T**	**F**	**T**	**F**	**F**	**T**
F	**F**	**F**	**F**	**T**	**F**	**F**	**F**

g.

						↓								
D	F	G	P	R	P	⊃	(F	⊃	[~	(D	∨	G)	&	R])
T	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**
T	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**
T	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**
T	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**
T	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**
T	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**
T	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
T	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
T	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**
T	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**
T	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**
T	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**
F	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**
F	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**
F	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**
F	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**
F	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
F	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**F**
F	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
F	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**F**
F	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**
F	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**
F	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**
F	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**
F	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
F	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**
F	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**
F	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**

Section 3.2E

1.a. Truth-functionally indeterminate

			↓	
A	~	A	⊃	A
T	**F**	**T**	**T**	**T**
F	**T**	**F**	**F**	**F**

c. Truth-functionally true

						↓					
A	(A	≡	~	A)	⊃	~	(A	≡	~	A)
T	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
F	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**F**

e. Truth-functionally indeterminate

							↓				
B	D	(~	B	&	~	D)	∨	~	(B	∨	D)
T	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**
T	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**
F	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**

g. Truth-functionally indeterminate

										↓				
A	B	C	[(A	∨	B)	&	(A	∨	C)]	⊃	~	(B	&	C)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
T	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**
F	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**
F	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**

i. Truth-functionally true

						↓					
J	K	(J	∨	~	K)	≡	~	~	(K	⊃	J)
T	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**
F	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**
F	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**F**

k. Truth-functionally true

											↓		
A	D	[(A	∨	~	D)	&	~	(A	&	D)]	⊃	~	D
T	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**
F	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**

2.a. Not truth-functionally true

					↓				
F	H	(F	∨	H)	∨	(~	F	≡	H)
F	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**

c. Truth-functionally true

					↓					
A	B	C	~	A	⊃	[(B	&	A)	⊃	C]
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**
T	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
T	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**
F	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**F**
F	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**
F	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**

e. Truth-functionally true

							↓	
C	[(C	∨	~	C)	⊃	C]	⊃	C
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**
F	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**

3.a. Truth-functionally false

					↓				
B	D	(B	≡	D)	&	(B	≡	~	D)
T	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**
T	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**
F	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**
F	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**

c. Not truth-functionally false

			↓			
A	B	A	≡	(B	≡	A)
T	**T**	**T**	**T**	**T**	**T**	**T**

e. Not truth-functionally false

							↓		
C	D	[(C	∨	D)	≡	C]	⊃	~	C
F	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**

4.a. False. For example, while '(A ⊃ A)' is truth-functionally true, '(A ⊃ A) & A' is not.

c. True. There cannot be any truth-value assignment on which the antecedent is true and the consequent false because there is no truth-value assignment on which the consequent is false.

e. False. For example, although '(A & ~ A)' is truth-functionally false, 'C ∨ (A & ~ A)' is not.

g. True. Since a sentence ~ **P** is false on a truth-value assignment if and only if **P** is true on the truth-value assignment, **P** is truth-functionally true if and only if ~ **P** is truth-functionally false.

i. False. For example, '(A ∨ ~ A)' is truth-functionally true, but '(A ∨ ~ A) ⊃ B' is truth-functionally indeterminate.

5.a. On every truth-value assignment, **P** is true and **Q** is false. Hence **P** ≡ **Q** is false on every truth-value assignment. Therefore **P** ≡ **Q** is truth-functionally false.

c. No. Both 'A' and '~ A' are truth-functionally indeterminate, but 'A ∨ ~ A' is truth-functionally true.

Section 3.3E

1.a. Not truth-functionally equivalent

		↓				↓			
A	B	~	(A	&	B)	~	(A	∨	B)
T	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**
F	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**

c. Truth-functionally equivalent

			↓				↓		
H	K	K	≡	H	~	K	≡	~	H
T	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**
T	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**
F	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**
F	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**

e. Truth-functionally equivalent

					↓							↓				
F	G	(G	⊃	F)	⊃	(F	⊃	G)	(G	≡	F)	∨	(~	F	∨	G)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**
T	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**
F	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**T**
F	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**

g. Not truth-functionally equivalent

							↓								↓		
H	J	K	~	(H	&	J)	≡	(J	≡	~	K)	(H	&	J)	⊃	~	K
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**
T	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**T**
T	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**
F	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**
F	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**
F	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**F**

i. Not truth-functionally equivalent

									↓									↓		
A	C	D	[A	∨	~	(D	&	C)]	⊃	~	D	[D	∨	~	(A	&	C)]	⊃	~	A
T	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**
T	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**
F	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**
F	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**

k. Not truth-functionally equivalent

				↓										↓	
F	G	H	F	∨	~	(G	∨	~	H)	(H	≡	~	F)	∨	G
T	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**
T	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**F**
F	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**T**
F	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**T**
F	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**

2.a. Truth-functionally equivalent

			↓				↓	
G	H	G	∨	H	~	G	⊃	H
T	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**
F	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**

c. Truth-functionally equivalent

					↓			↓	
A	D	(D	≡	A)	&	D	D	&	A
T	T	T	T	T	T	T	T	T	T
T	F	F	F	T	F	F	F	F	T
F	T	T	F	F	F	T	T	F	F
F	F	F	T	F	F	F	F	F	F

e. Not truth-functionally equivalent

		↓					↓				
A	A	≡	(~	A	≡	A)	~	(A	⊃	~	A)
T	T	F	F	T	F	T	T	T	F	F	T

3.a. Not truth-functionally equivalent

C: The sky clouds over.
N: The night will be clear.
M: The moon will shine brightly.

				↓					↓				
C	M	N	C	∨	(N	&	M)	M	≡	(N	&	~	C)
T	T	T	T	T	T	T	T	T	F	T	F	F	T
T	T	F	T	T	F	F	T	T	F	F	F	F	T
T	F	T	T	T	T	F	F	F	T	T	F	F	T
T	F	F	T	T	F	F	F	F	T	F	F	F	T
F	T	T	F	T	T	T	T	T	T	T	T	T	F
F	T	F	F	F	F	F	T	T	F	F	F	T	F
F	F	T	F	F	T	F	F	F	F	T	T	T	F
F	F	F	F	F	F	F	F	F	T	F	F	T	F

c. Truth-functionally equivalent

D: The *Daily Herald* reports on our antics.
A: Our antics are effective.

			↓				↓		
A	D	D	⊃	A	~	A	⊃	~	D
T	T	T	T	T	F	T	T	F	T
T	F	F	T	T	F	T	T	T	F
F	T	T	F	F	T	F	F	F	T
F	F	F	T	F	T	F	T	T	F

e. Not truth-functionally equivalent

M: Mary met Tom.
L: Mary liked Tom.
G: Mary asked George to the movies.

						↓							↓	
G	L	M	(M	&	L)	⊃	~	G	(M	&	~	L)	⊃	G
T	T	T	T	T	T	F	F	T	T	F	F	T	T	T
T	T	F	F	F	T	T	F	T	F	F	F	T	T	T
T	F	T	T	F	F	T	F	T	T	T	T	F	T	T
T	F	F	F	F	F	T	F	T	F	F	T	F	T	T
F	T	T	T	T	T	T	T	F	T	F	F	T	T	F
F	T	F	F	F	T	T	T	F	F	F	F	T	T	F
F	F	T	T	F	F	T	T	F	T	T	T	F	F	F
F	F	F	F	F	F	T	T	F	F	F	T	F	T	F

4.a. Yes. **P** and **Q** have the same truth-value on every truth-value assignment. On every truth-value assignment on which they are both true, ~ **P** and ~ **Q** are both false, and on every truth-value assignment on which they are both false, ~ **P** and ~ **Q** are both true. It follows that ~ **P** and ~ **Q** are truth-functionally equivalent.

c. If **P** and **Q** are truth-functionally equivalent then they have the same truth-value on every truth-value assignment. On those assignments on which they are both true, the second disjunct of ~ **P** ∨ **Q** is true and so is the disjunction. On those assignments on which they are both false, the first disjunct of ~ **P** ∨ **Q** is true and so is the disjunction. So ~ **P** ∨ **Q** is true on every truth-value assignment.

Section 3.4E

1.a. Truth-functionally consistent

				↓			↓			↓	
A	B	C	A	⊃	B	B	⊃	C	A	⊃	C
T	T	T	T	T	T	T	T	T	T	T	T
T	T	F	T	T	T	T	F	F	T	F	F
T	F	T	T	F	F	F	T	T	T	T	T
T	F	F	T	F	F	F	T	F	T	F	F
F	T	T	F	T	T	T	T	T	F	T	T
F	T	F	F	T	T	T	F	F	F	T	F
F	F	T	F	T	F	F	T	T	F	T	T
F	F	F	F	T	F	F	T	F	F	T	F

c. Truth-functionally inconsistent

			↓							↓							↓			
H	J	L	~	[J	∨	(H	⊃	L)]	L	≡	(~	J	∨	~	H)	H	≡	(J	∨	L)
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**
T	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**
T	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**
F	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**
F	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**
F	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**F**

e. Truth-functionally inconsistent

					↓		↓		↓	
H	J	(J	⊃	J)	⊃	H	~	J	~	H
T	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**
T	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**
F	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**

g. Truth-functionally consistent

			↓	↓	↓
A	B	C	A	B	C
T	**T**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**F**
T	**F**	**T**	**T**	**F**	**T**
T	**F**	**F**	**T**	**F**	**F**
F	**T**	**T**	**F**	**T**	**T**
F	**T**	**F**	**F**	**T**	**F**
F	**F**	**T**	**F**	**F**	**T**
F	**F**	**F**	**F**	**F**	**F**

i. Truth-functionally consistent

						↓				↓		↓	
A	B	C	(A	&	B)	∨	(C	⊃	B)	~	A	~	B
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**
T	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
T	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
F	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**
F	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**
F	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**

2.a. Truth-functionally consistent

				↓						↓	
B	D	E	B	⊃	(D	⊃	E)	~	D	&	B
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**

c. Truth-functionally consistent

				↓					↓		
F	J	K	F	⊃	(J	∨	K)	F	≡	~	J
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**

e. Truth-functionally consistent

					↓					↓
A	B	(A	⊃	B)	≡	(~	B	∨	B)	A
T	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**

3.a. Truth-functionally inconsistent

S: Space is infinitely divisible.
Z: Zeno's paradoxes are compelling.
C: Zeno's paradoxes are convincing.

				↓		↓				↓
C	S	Z	S	⊃	Z	~	(C	∨	Z)	S
T	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**F**
T	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**
F	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**
F	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**T**
F	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**
F	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**

c. Truth-functionally consistent

E: Eugene O'Neill was an alcoholic.
P: Eugene O'Neill's plays show that he was an alcoholic.
I: *The Iceman Cometh* must have been written by a teetotaler.
F: Eugene O'Neill was a fake.

				↓	↓	↓		↓	
E	F	I	P	E	P	I	E	∨	F
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**
T	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**
T	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**
T	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**
T	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**
T	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**
F	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**T**
F	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**
F	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**T**
F	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**
F	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**
F	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**

e. Truth-functionally consistent

R: The Red Sox will win next Sunday.
J: Joan bet $5.00.
E: Joan will buy Ed a hamburger.

				↓						↓		
E	J	R	R	⊃	(J	⊃	E)	~	R	&	~	E
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**
T	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**
T	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**
T	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**
F	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**F**
F	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**
F	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**

4.a. First assume that {**P**} is truth-functionally inconsistent. Then, since **P** is the only member of {**P**}, there is no truth-value assignment on which **P** is true; so **P** is false on every truth-value assignment. But then ~ **P** is true on every truth-value assignment, and so ~ **P** is truth-functionally true.

Now assume that ~ **P** is truth-functionally true. Then ~ **P** is true on every truth-value assignment, and so **P** is false on every truth-value assignment. But then there is no truth-value assignment on which **P**, the only member of {**P**}, is true, and so the set is truth-functionally inconsistent.

c. No. For example, 'A' and '~ A' are both truth-functionally indeterminate, but {A, ~ A} is truth-functionally inconsistent.

Section 3.5E

1.a. Truth-functionally valid

				↓					↓			↓		↓	
A	H	J	A	⊃	(H	&	J)	J	≡	H	~	J	~	A
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**
T	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**
T	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**T**
F	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**
F	**F**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**

c. Truth-functionally valid

							↓										↓				↓		
A	D	G	(D	≡	~	G)	&	G	(G	∨	[(A	⊃	D)	&	A])	⊃	~	D	G	⊃	~	D
T	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**
T	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**
T	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**
T	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**F**
F	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**
F	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**T**
F	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**F**
F	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**F**

e. Truth-functionally valid

						↓				↓		↓	
C	D	E	(C	⊃	D)	⊃	(D	⊃	E)	D	C	⊃	E
T	T	T	T	T	T	T	T	T	T	T	T	T	T
T	T	F	T	T	T	F	T	F	F	T	T	F	F
T	F	T	T	F	F	T	F	T	T	F	T	T	T
T	F	F	T	F	F	T	F	T	F	F	T	F	F
F	T	T	F	T	T	T	T	T	T	T	F	T	T
F	T	F	F	T	T	F	T	F	F	T	F	T	F
F	F	T	F	T	F	T	F	T	T	F	F	T	T
F	F	F	F	T	F	T	F	T	F	F	F	T	F

g. Truth-functionally valid

					↓										↓				
G	H	(G	≡	H)	∨	(~	G	≡	H)	(~	G	≡	~	H)	∨	~	(G	≡	H)
T	T	T	T	T	T	F	T	F	T	F	T	T	F	T	T	F	T	T	T
T	F	T	F	F	T	F	T	T	F	F	T	F	T	F	T	T	T	F	F
F	T	F	F	T	T	T	F	T	T	T	F	F	F	T	T	T	F	F	T
F	F	F	T	F	T	T	F	F	F	T	F	T	T	F	T	F	F	T	F

i. Truth-functionally invalid

					↓						↓				↓	
F	G	~	~	F	⊃	~	~	G	~	G	⊃	~	F	G	⊃	F
T	T	T	F	T	T	T	F	T	F	T	T	F	T	T	T	T
T	F	T	F	T	F	F	T	F	T	F	F	F	T	F	T	T
F	T	F	T	F	T	T	F	T	F	T	T	T	F	T	F	F
F	F	F	T	F	T	F	T	F	T	F	T	T	F	F	T	F

2.a. Truth-functionally valid

					↓						↓					↓	
J	M	(J	∨	M)	⊃	~	(J	&	M)	M	≡	(M	⊃	J)	M	⊃	J
T	T	T	T	T	F	F	T	T	T	T	T	T	T	T	T	T	T
T	F	T	T	F	T	T	T	F	F	F	F	F	T	T	F	T	T
F	T	F	T	T	T	T	F	F	T	T	F	T	F	F	T	F	F
F	F	F	F	F	T	T	F	F	F	F	F	F	T	F	F	T	F

c. Truth-functionally valid

			↓						↓			↓		
A	B	A	⊃	~	A	(B	⊃	A)	⊃	B	A	≡	~	B
T	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**
T	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**T**
F	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**

e. Truth-functionally invalid

				↓										↓				↓	
A	B	C	A	&	~	[(B	&	C)	≡	(C	⊃	A)]	B	⊃	~	B	~	C	⊃	C
T	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**F**

3.a. Truth-functionally valid

					↓			
B	C	(B	&	C)	⊃	(B	∨	C)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**
T	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**
F	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**T**
F	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**

c. Truth-functionally invalid

													↓					
J	T	([(J	⊃	T)	⊃	J]	&	[(T	⊃	J)	⊃	T])	⊃	(~	J	∨	~	T)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**

e. Truth-functionally invalid

										↓	
B	C	D	[(B	&	C)	&	(B	∨	D)]	⊃	D
T	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**

4.a. Truth-functionally invalid

S: ‘Stern’ means the same as ‘star’.
N: ‘Nacht’ means the same as ‘day’.

			↓		↓		↓	
N	S	N	⊃	S	~	N	~	S
T	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**
T	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**
F	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**F**

c. Truth-functionally valid

S: September has 30 days.
A: April has 30 days.
N: November has 30 days.
F: February has 40 days.
M: May has 30 days.

						↓								↓				↓
A	F	M	N	S	S	&	(A	&	N)	(A	≡	~	M)	&	(N	⊃	M)	F
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**T**
T	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**T**
T	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**
T	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**
T	**T**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**
T	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**T**
T	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**T**
T	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**
T	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**
T	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**F**
T	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**F**
T	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**F**
T	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**F**
F	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**
F	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**
F	**T**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**
F	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**
F	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**
F	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**
F	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**T**
F	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**
F	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**F**
F	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**
F	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**
F	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**
F	**F**	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**
F	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**
F	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**

e. Truth-functionally valid

D: Computers can have desires.
E: Computers can have emotions.
T: Computers can think.

				↓			↓			↓			↓	
D	E	T	T	≡	E	E	⊃	D	D	⊃	~	T	~	T
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**
T	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**T**
T	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**F**
F	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**	**T**
F	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**	**T**	**F**
F	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**
F	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**

5.a. Suppose that the argument is truth-functionally valid. Then there is no truth-value assignment on which $\mathbf{P}_1, \ldots, \mathbf{P}_{\mathbf{n}}$ are all true and $\mathbf{Q}$ is false. But, by the characteristic truth-table for '&', the iterated conjunction $(\ldots(\mathbf{P}_1 \,\&\, \mathbf{P}_2) \,\&\, \ldots \mathbf{P}_{\mathbf{n}})$ has the truth-value **T** on a truth-value assignment if and only if all of $\mathbf{P}_1, \ldots, \mathbf{P}_{\mathbf{n}}$ have the truth-value **T** on that assignment. So, on our assumption, there is no truth-value assignment on which the antecedent of $(\ldots(\mathbf{P}_1 \,\&\, \mathbf{P}_2) \,\&\, \ldots \,\&\, \mathbf{P}_{\mathbf{n}}) \supset \mathbf{Q}$ has the truth-value **T** and the consequent has the truth-value **F**. It follows that there is no truth-value assignment on which the corresponding material conditional is false, so it is truth-functionally true.

Assume that $(\ldots(\mathbf{P}_1 \,\&\, \mathbf{P}_2) \,\&\, \ldots \,\&\, \mathbf{P}_{\mathbf{n}}) \supset \mathbf{Q}$ is truth-functionally true. Then there is no truth-value assignment on which the antecedent is true and the consequent false. But the iterated conjunction is true if and only if the sentences $\mathbf{P}_1, \ldots, \mathbf{P}_{\mathbf{n}}$ are all true. So there is no truth-value assignment on which $\mathbf{P}_1, \ldots, \mathbf{P}_{\mathbf{n}}$ are all true and $\mathbf{Q}$ is false; hence the argument is truth-functionally valid.

c. No. For example, {A ⊃ B} ⊨ '~ A ∨ B'. But {A ⊃ B} does not entail '~ A', nor does it entail 'B'.

Section 3.6E

1.a. If {~ **P**} is truth-functionally inconsistent, then there is no truth-value assignment on which ~ **P** is true (since ~ **P** is the only member of its unit set). But then ~ **P** is false on every truth-value assignment, so **P** is true on every truth-value assignment and is truth-functionally true.

c. If $\Gamma \cup$ {~ **P**} is truth-functionally inconsistent, then there is no truth-value assignment on which every member of $\Gamma \cup$ {~ **P**} is true. But ~ **P** is true on a truth-value assignment if and only if **P** is false on that assignment. Hence there is no truth-value assignment on which every member of Γ is true and **P** is false. Hence Γ ⊨ **P**.

2.a. **P** is truth-functionally true if and only if the set {~ **P**} is truth-functionally inconsistent. But {~**P**} is the same set as ∅ ∪ {~ **P**}. So **P** is truth-functionally true if and only if ∅ ∪ {~ **P**} is truth-functionally inconsistent. But we have already seen, by previous results, that ∅ ∪ {~ **P**} is truth-functionally inconsistent if and only if ∅ ⊨ **P**. Hence **P** is truth-functionally true if and only if ∅ ⊨ **P**.

c. Assume that Γ is truth-functionally inconsistent. Then there is no truth-value assignment on which every member of Γ is true. Let **P** be an *arbitrarily* selected sentence of *SL*. Then there is no truth-value assignment on which every member of Γ is true and **P** false since there is no truth-value assignment on which every member of Γ is true. Hence Γ ⊨ **P**.

3.a. Let Γ be a truth-functionally consistent set. Then there is at least one truth-value assignment on which every member of Γ is true. But **P** is also true on such an assignment since a truth-functionally true sentence is true on every truth-value assignment. Hence on at least one truth-value assignment every member of Γ ∪ {**P**} is true; so the set is truth-functionally consistent.

4.a. **P** is either true or false on each truth-value assignment. On any assignment on which **P** is true, **Q** is true (because {**P**} ⊨ **Q**) and so **Q** ∨ **R** is true. On any assignment on which **P** is false, ~ **P** is true, **R** is therefore also true (because {~ **P**} ⊨ **R**), and so **Q** ∨ **R** is true as well. Either way, then, **Q** ∨ **R** is true—so the sentence is truth-functionally true.

c. Assume that every member of Γ ∪ Γ′ is true on some truth-value assignment. Then every member of Γ is true, and so **P** is true (because Γ ⊨ **P**). Every member of Γ′ is also true, and so **Q** is true (because Γ′ ⊨ **Q**). Therefore **P** & **Q** is true. So Γ ∪ Γ′ ⊨ **P** & **Q**.

Section 4.2E

1. a.

1.	$A \& \sim (B \vee A)$ ✓	SM
2.	A	1 &D
3.	$\sim (B \vee A)$ ✓	1 &D
4.	$\sim B$	3 ~ ∨D
5.	$\sim A$	3 ~ ∨D
	×	

Since the truth-tree is closed, the set is truth-functionally inconsistent.

c.

1.	$\sim (A \vee B) \& (A \vee \sim B)$ ✓		SM
2.	$\sim (A \vee B)$ ✓		1 &D
3.	$A \vee \sim B$ ✓		1 &D
4.	$\sim A$		2 ~ ∨D
5.	$\sim B$		2 ~ ∨D
6.	A	$\sim B$	2 ∨D
	×	o	

Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

A	B
F	**F**

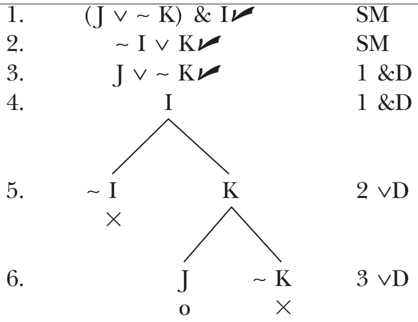
e.

1.	$(A \vee B) \& (A \vee \sim B)$ ✓				SM
2.	$A \vee B$ ✓				1 &D
3.	$A \vee \sim B$ ✓				1 &D
4.	A		B		2 ∨D
5.	A	$\sim B$	A	$\sim B$	3 ∨D
	o	o	o	×	

Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

A	B
T	**T**
T	**F**

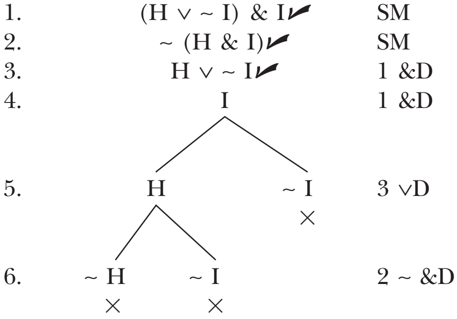
g.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

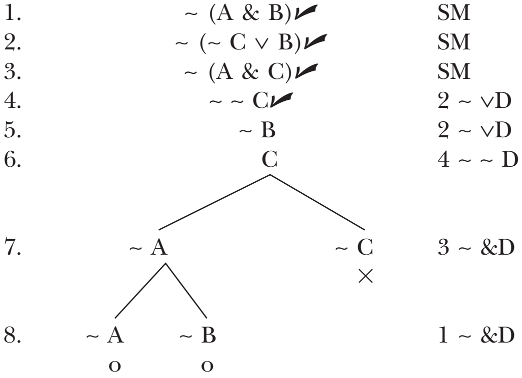
I	J	K
T	**T**	**T**

i.



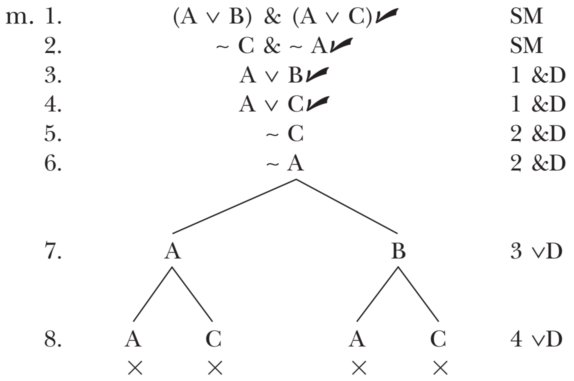
Since the truth-tree is closed, the set is truth-functionally inconsistent.

k.

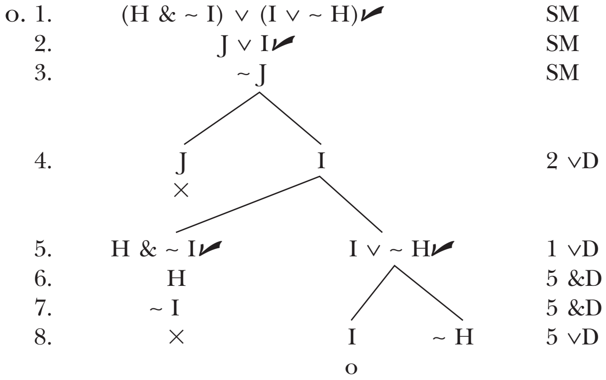


Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

A	B	C
F	**F**	**T**



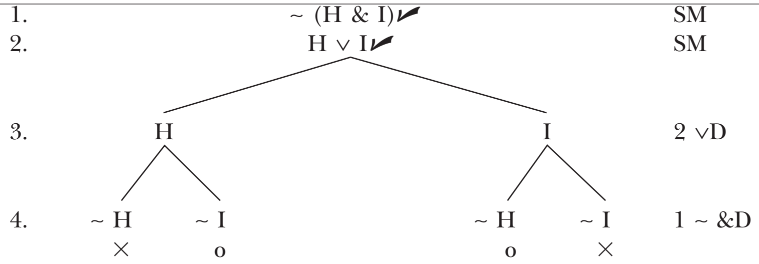
Since the truth-tree is closed, the set is truth-functionally inconsistent.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

H	I	J
T	**T**	**F**
F	**T**	**F**

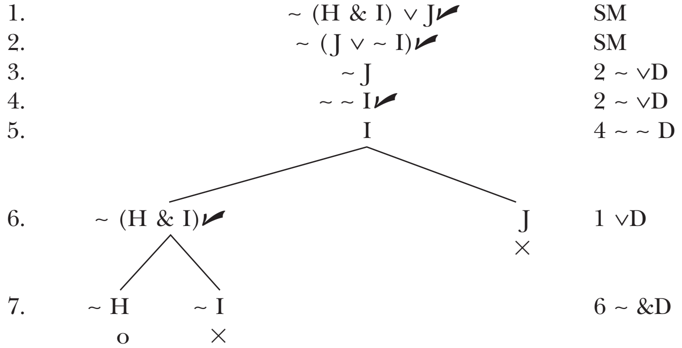
2. a.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

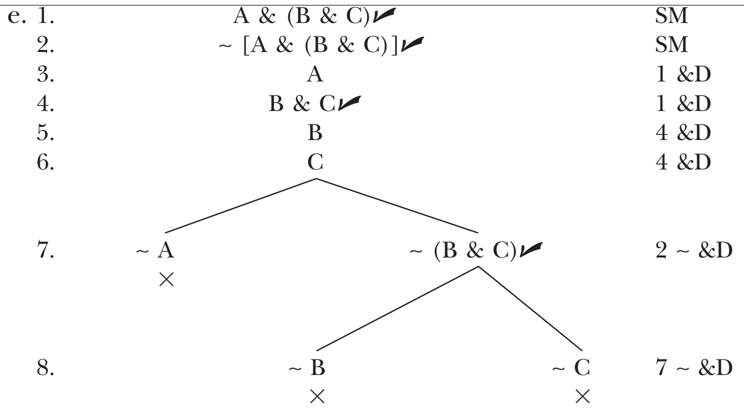
H	I
T	**F**
F	**T**

c.

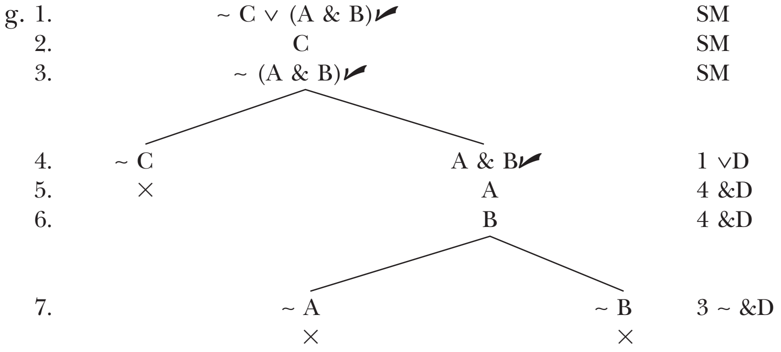


Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments is

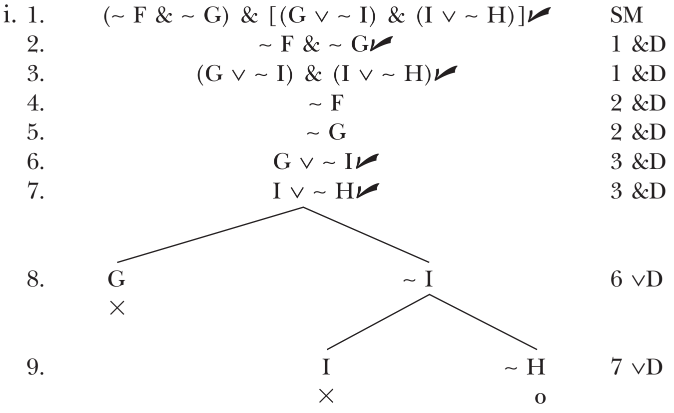
H	I	J
F	**T**	**F**



Since the truth-tree is closed, the set is truth-functionally inconsistent.



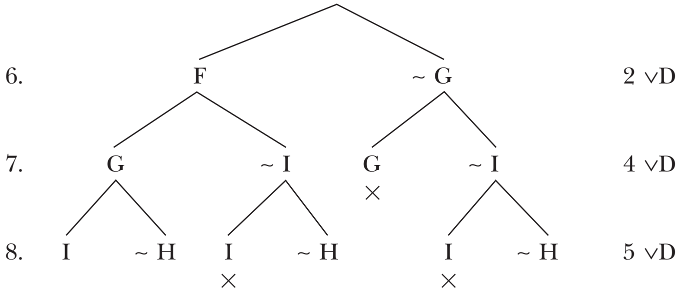
Since the truth-tree is closed, the set is truth-functionally inconsistent.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

F	G	H	I
F	**F**	**F**	**F**

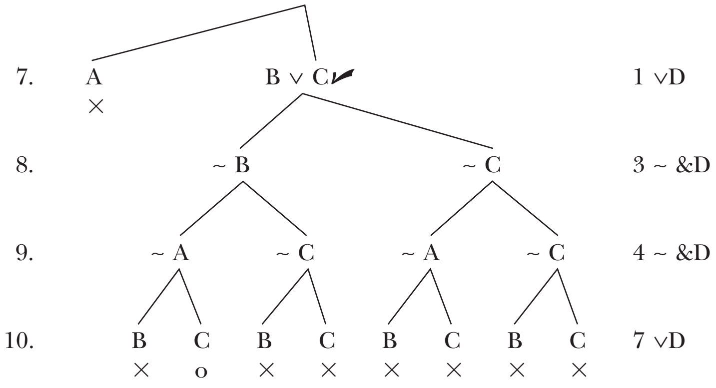
k. 1. $(F \vee \sim G) \,\&\, [(G \vee \sim I) \,\&\, (I \vee \sim H)]$ ✓ SM
2. $F \vee \sim G$ ✓ 1 &D
3. $(G \vee \sim I) \,\&\, (I \vee \sim H)$ ✓ 1 &D
4. $G \vee \sim I$ ✓ 3 &D
5. $I \vee \sim H$ ✓ 3 &D



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

F	G	H	I
T	**T**	**T**	**T**
T	**T**	**F**	**T**
T	**T**	**F**	**F**
T	**F**	**F**	**F**
F	**F**	**F**	**F**

m. 1. A ∨ (B ∨ C)✓ SM
2. ~ (A ∨ B)✓ SM
3. ~ (B & C)✓ SM
4. ~ (A & C)✓ SM
5. ~ A 2 ~ ∨D
6. ~ B 2 ~ ∨D



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

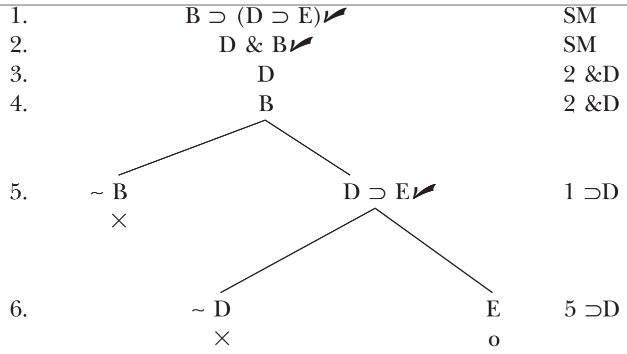
A	B	C
F	**F**	**T**

Section 4.3.E

1. a. 1. ~ (A ⊃ B)✓ SM
2. ~ (B ⊃ A)✓ SM
3. A 1 ~ ⊃D
4. ~ B 1 ~ ⊃D
5. B 2 ~ ⊃D
6. ~ A 2 ~ ⊃D
×

Since the truth-tree is closed, the set is truth-functionally inconsistent.

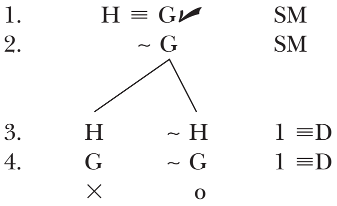
c.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

B	D	E
T	**T**	**T**

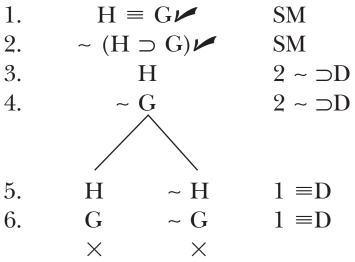
e.



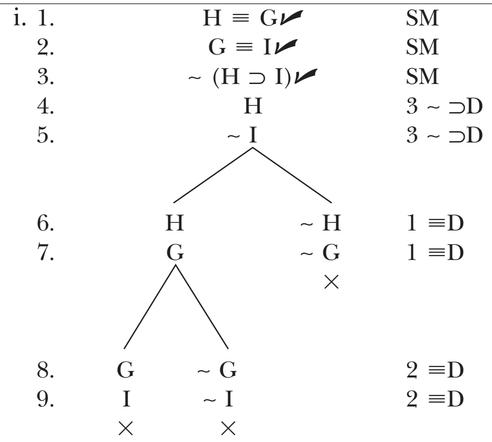
Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

H	G
F	**F**

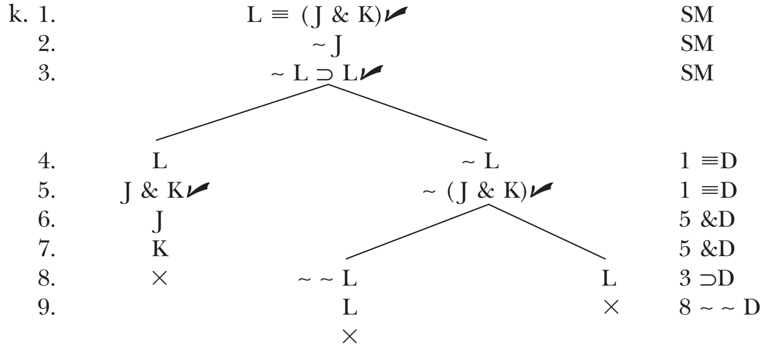
g.



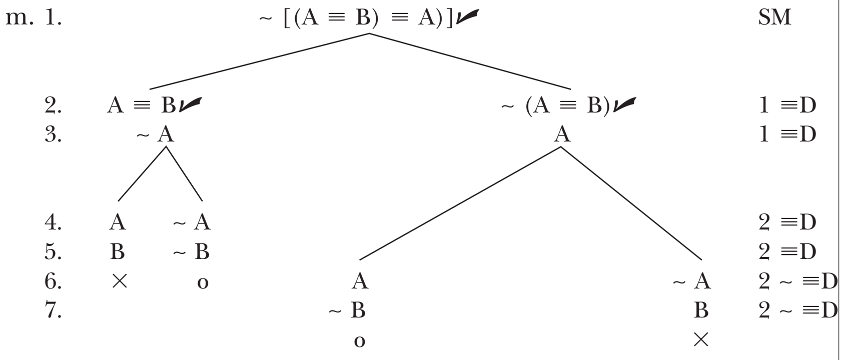
Since the truth-tree is closed, the set is truth-functionally inconsistent.



Since the truth-tree is closed, the set is truth-functionally inconsistent.



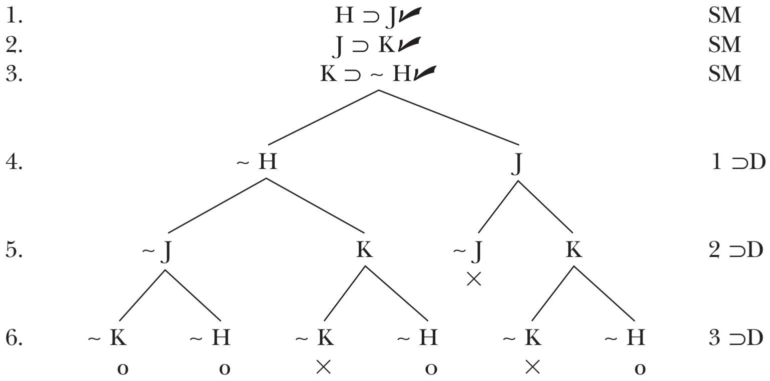
Since the truth-tree is closed, the set is truth-functionally inconsistent.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

A	B
T	**F**
F	**F**

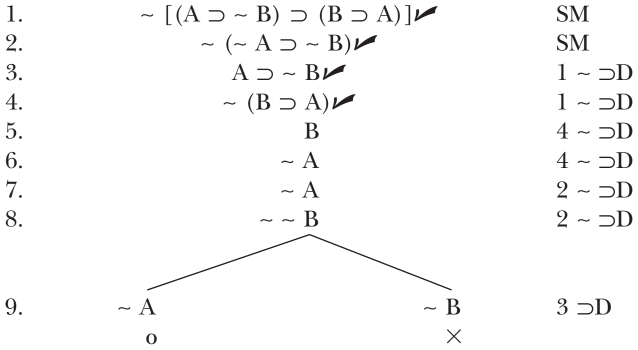
o.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

H	J	K
F	**F**	**F**
F	**T**	**T**
F	**F**	**T**

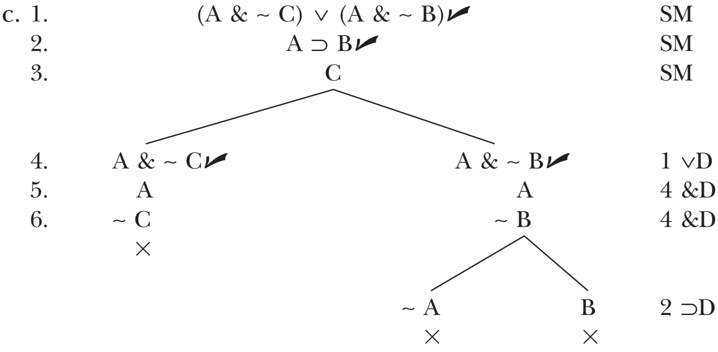
2. a.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable fragment is

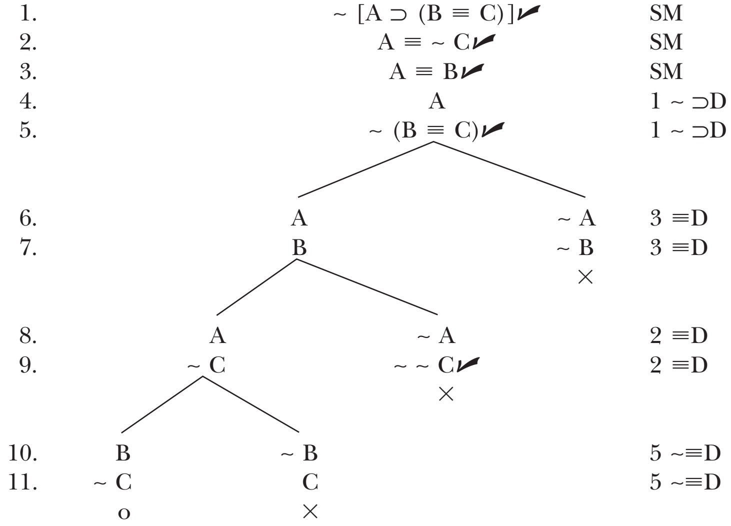
A	B
F	**T**

c.



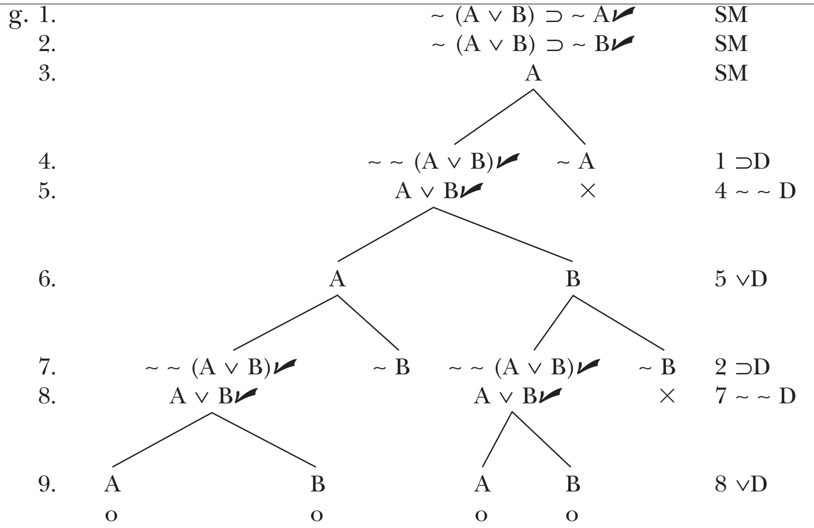
Since the truth-tree is closed, the set is truth-functionally inconsistent.

e.



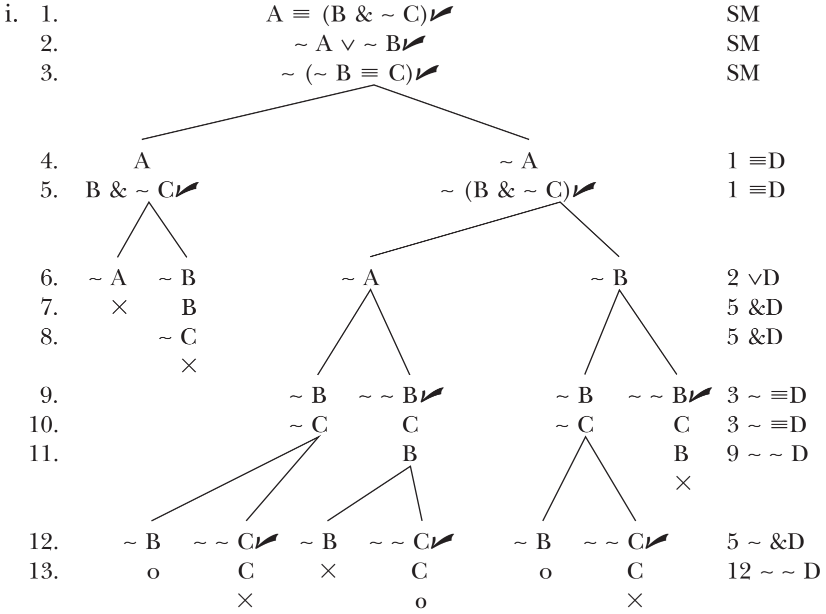
Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

A	B	C
T	**T**	**F**



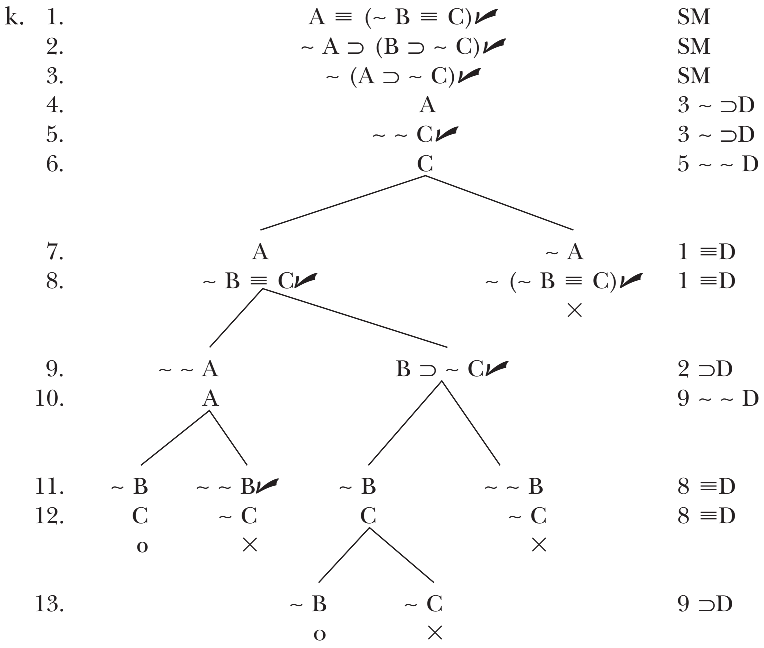
Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

A	B
T	**T**
T	**F**



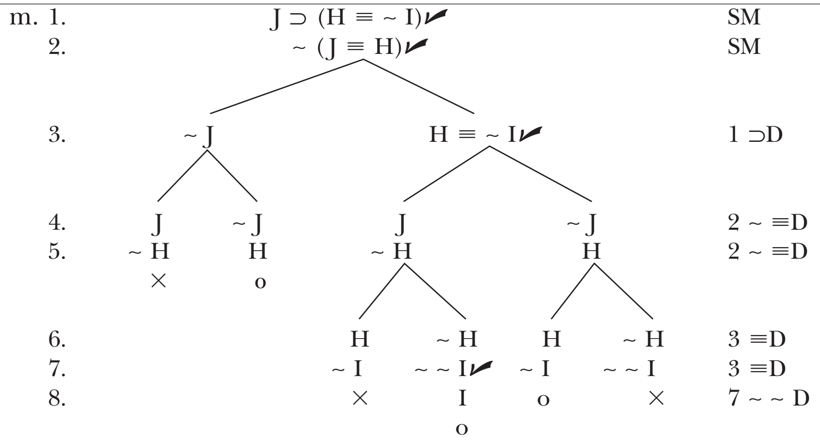
Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

A	B	C
F	**F**	**F**
F	**F**	**T**
F	**T**	**T**



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

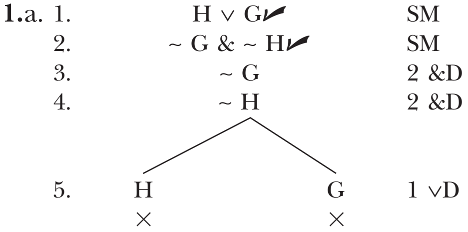
A	B	C
T	**F**	**T**



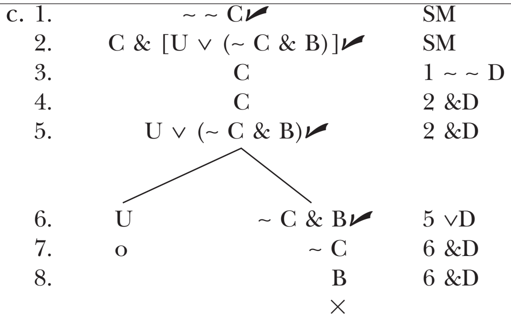
Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable set of truth-value assignments is

J	H	I
F	**T**	**T**
F	**T**	**F**
T	**F**	**T**

Section 4.4E

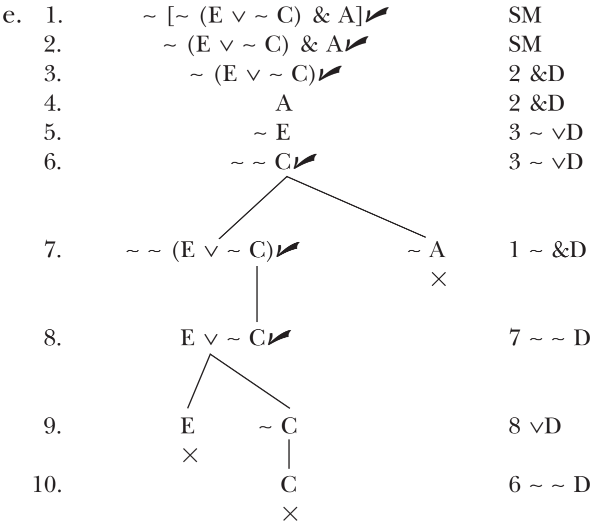


Since the truth-tree is closed, the set is truth-functionally inconsistent.

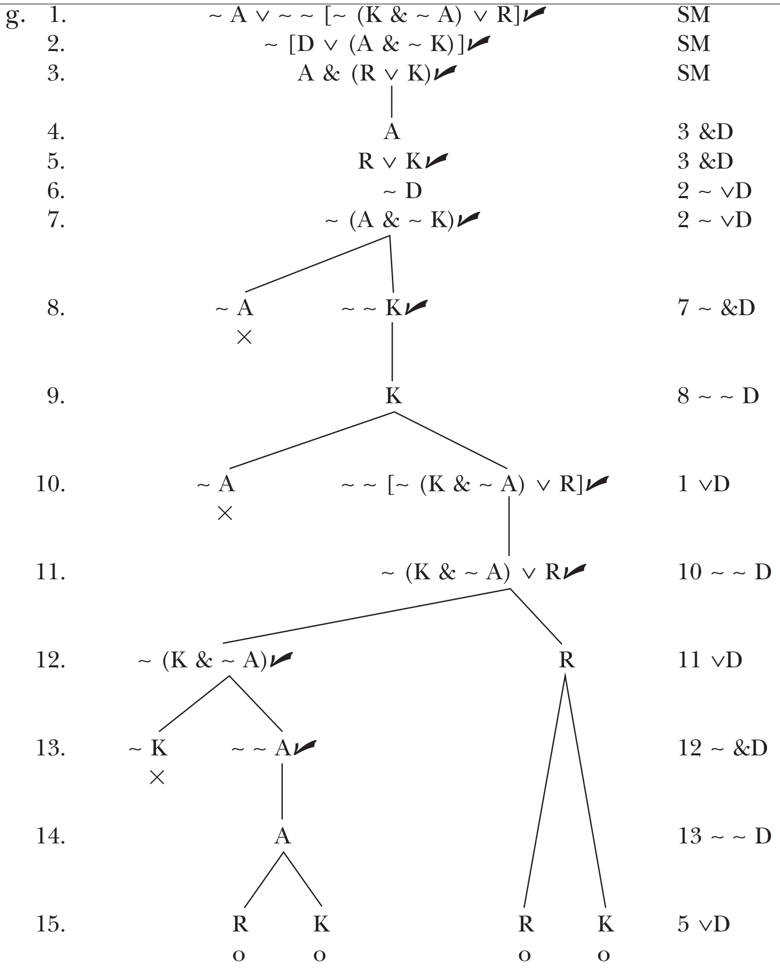


Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

B	C	U
F	**T**	**T**
T	**T**	**T**



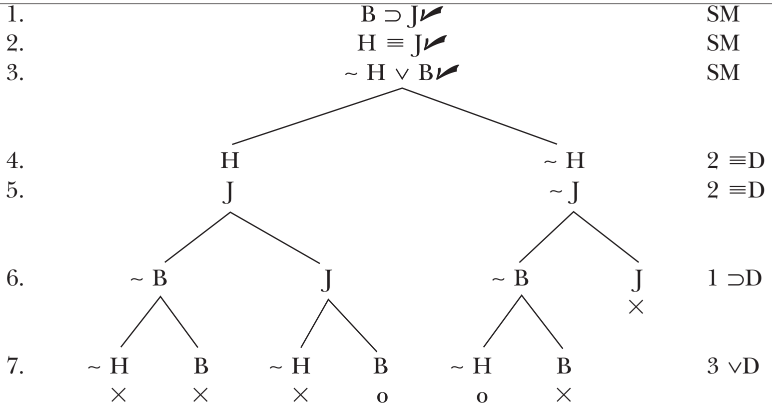
Since the truth-tree is closed, the set is truth-functionally inconsistent.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

A	D	K	R
T	**F**	**T**	**T**
T	**F**	**T**	**F**

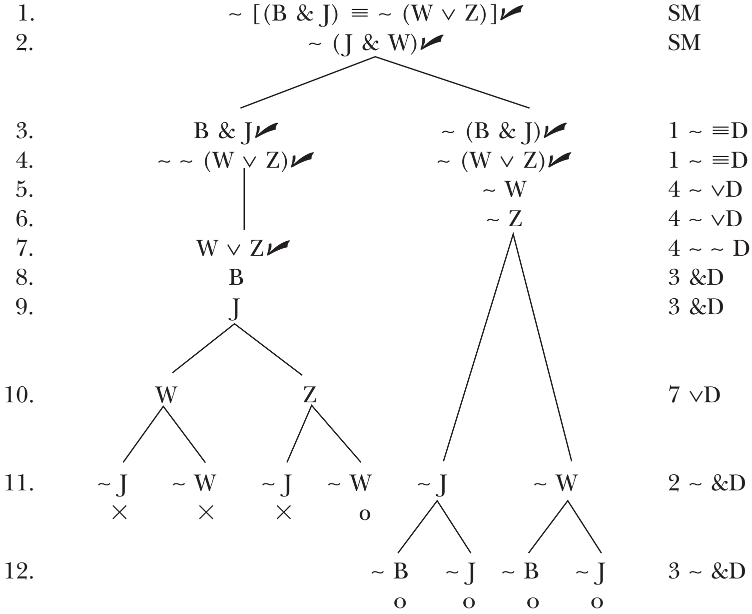
i.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

B	H	J
T	**T**	**T**
F	**F**	**F**

k.



Since the truth-tree has at least one completed open branch, the set is truth-functionally consistent. The recoverable sets of truth-value assignments are

B	J	W	Z
T	**T**	**F**	**T**
T	**F**	**F**	**F**
F	**T**	**F**	**F**
F	**F**	**F**	**F**

2.a. True. Truth-trees test for consistency. A completed open branch shows that the set is consistent because it yields at least one truth-value assignment on which all the members of the set being tested are true. An open branch on a completed truth-tree is a completed open branch.

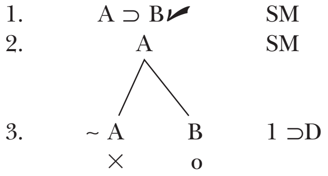
c. True. If a tree has a completed open branch, then we can recover from that branch a truth-value assignment on which every member of the set is true. And a set is, by definition, consistent if and only if there is at least one truth-value assignment on which all its members are true.

e. True. If all the branches are closed, there is no truth-value assignment on which all the members of the set being tested are true, and if there is no such assignment, that set is truth-functionally inconsistent.

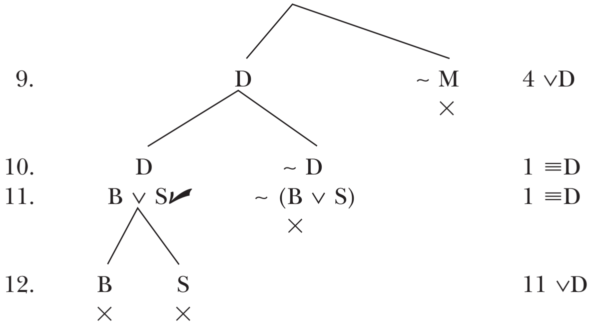
g. False. The number of branches on a completed tree and the number of distinct atomic components of the members of the set being tested are not related.

i. False. Closed branches represent unsuccessful attempts to find truth-value assignments on which all the members of the set being tested are true. No sets of truth-value assignments are recoverable from them; hence they do not yield assignments on which all the members of the set being tested are false.

k. False. The truth-tree for {A ⊃ B, A} has a closed branch.

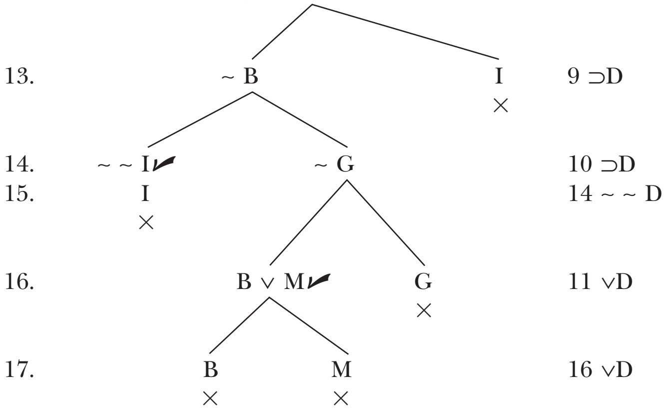


3.a. 1. $D \equiv (B \vee S)$ ✓ SM
2. $\sim (B \vee S)\ \&\ M$ ✓ SM
3. $N \supset M$ SM
4. $D \vee \sim M$ ✓ SM
5. $\sim (B \vee S)$ ✓ 2 &D
6. M 2 &D
7. $\sim B$ 5 ~ ∨D
8. $\sim S$ 5 ~ ∨D



Since the truth-tree is closed, the set is truth-functionally inconsistent.

c. 1. $[(B \vee M) \vee G]\ \&\ \sim [(B\ \&\ M)\ \&\ G]$ ✓ SM
2. $(B \supset I)\ \&\ (\sim I \supset \sim G)$ ✓ SM
3. $[U \supset (B \vee M)]\ \&\ \sim M$ ✓ SM
4. $U\ \&\ \sim I$ ✓ SM
5. U 4 &D
6. $\sim I$ 4 &D
7. $U \supset (B \vee M)$ 3 &D
8. $\sim M$ 3 &D
9. $B \supset I$ ✓ 2 &D
10. $\sim I \supset \sim G$ ✓ 2 &D
11. $(B \vee M) \vee G$ ✓ 1 &D
12. $\sim [(B\ \&\ M)\ \&\ G]$ 1 &D



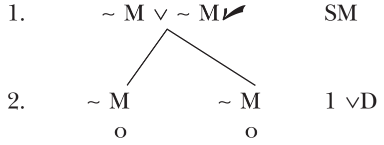
Since the truth-tree is closed, the set is truth-functionally inconsistent.

Section 4.5E

1.a. 1.	M & ~ M✓	SM
2.	M	1 &D
3.	~ M	1 &D
	×	

Since the truth-tree for the given sentence is closed, that sentence is truth-functionally false.

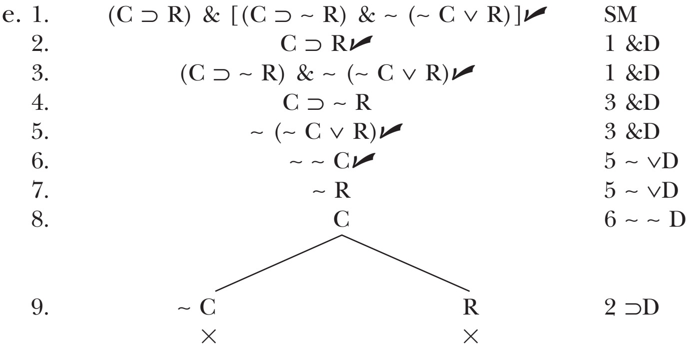
c.



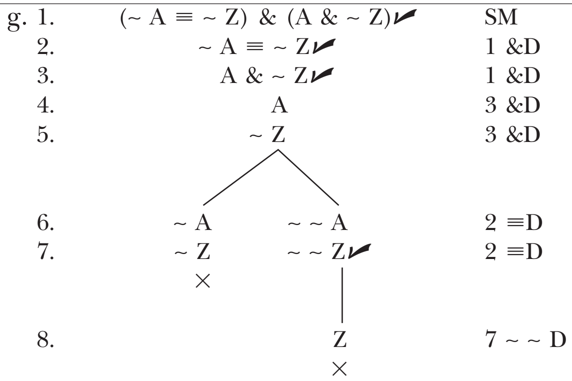
1.	~ (~ M ∨ ~ M)✓	SM
2.	~ ~ M✓	1 ~ ∨D
3.	~ ~ M✓	1 ~ ∨D
4.	M	2 ~ ~ D
5.	M	3 ~ ~ D
	o	

Neither the truth-tree for the given sentence nor the truth-tree for the negation of that sentence is closed, therefore the given sentence is truth-functionally indeterminate.

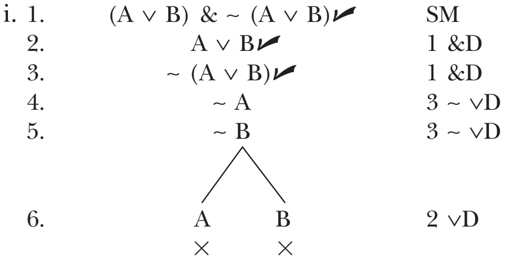
e.



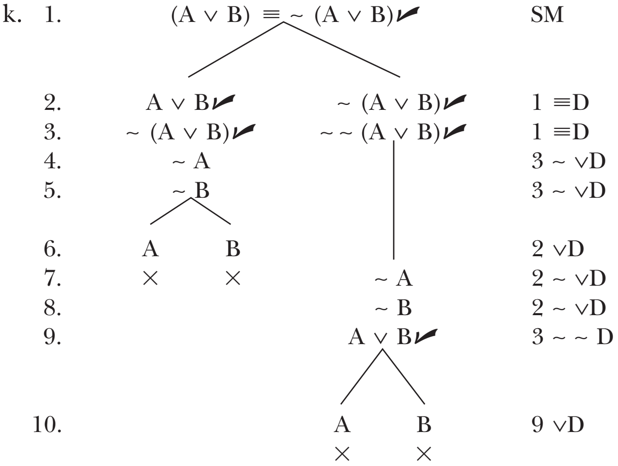
Since the truth-tree is closed, the sentence we are testing is truth-functionally false.



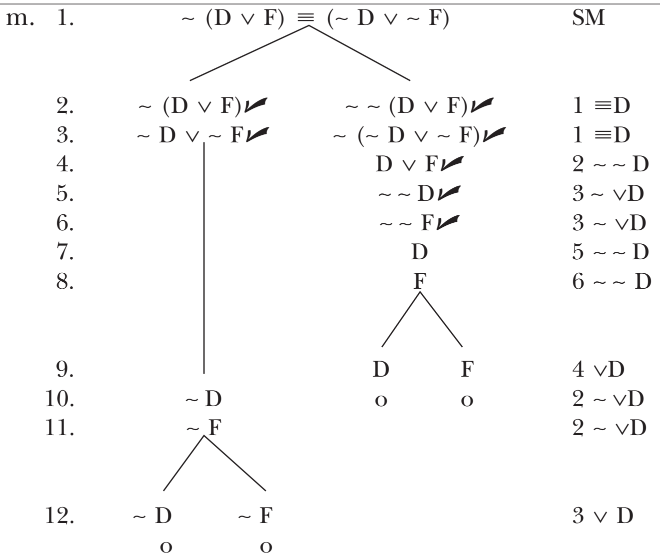
Since the truth-tree is closed, the sentence we are testing is truth-functionally false.

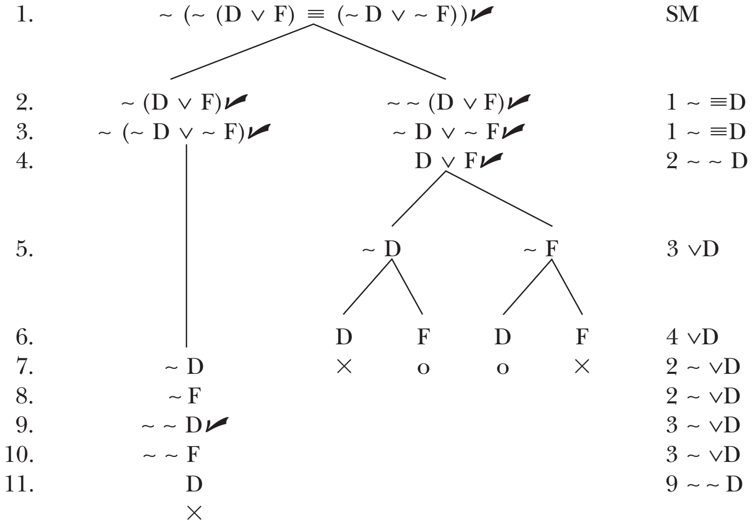


The tree is closed, so the sentence is truth-functionally false.



The tree is closed, so the sentence is truth-functionally false.





Neither the tree for the sentence nor the tree for its negation is closed. Therefore the sentence is truth-functionally indeterminate.

2.a.	1.	$\sim [(B \supset L) \vee (L \supset B)]$✓	SM
	2.	$\sim (B \supset L)$✓	$1 \sim \vee D$
	3.	$\sim (L \supset B)$✓	$1 \sim \vee D$
	4.	B	$2 \sim \supset D$
	5.	$\sim L$	$2 \sim \supset D$
	6.	L	$3 \sim \supset D$
	7.	$\sim B$	$3 \sim \supset D$
		×	

Since the truth-tree for the negation of the given sentence is closed, the given sentence is truth-functionally true.

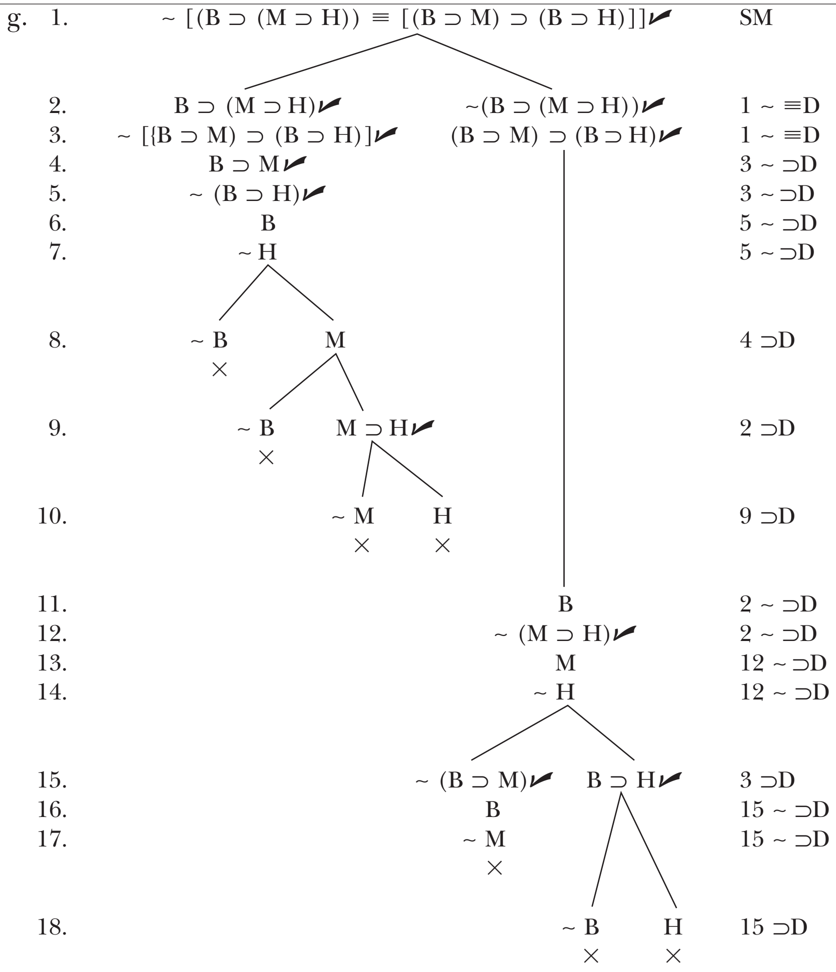
c.	1.	$\sim [(A \equiv K) \supset (A \vee K)]$✓		SM
	2.	$A \equiv K$✓		$1 \sim \vee D$
	3.	$\sim (A \vee K)$✓		$1 \sim \vee D$
	4.	$\sim A$		$3 \sim \vee D$
	5.	$\sim K$		$3 \sim \vee D$
	6.	A	$\sim A$	$2 \equiv D$
	7.	K	$\sim K$	$2 \equiv D$
		×	o	

Since the truth-tree for the negation of the given sentence is not closed, the given sentence is not truth-functionally true. The recoverable set of truth-value assignments is

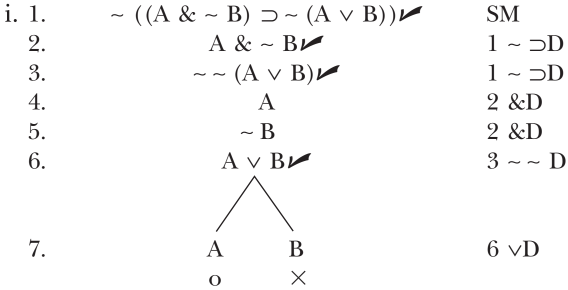
A	K
F	**F**

e.	1.	$\sim [[(J \supset Z) \ \& \sim Z] \supset \sim J]$✓		SM
	2.	$(J \supset Z) \ \& \sim Z$✓		$1 \sim \vee D$
	3.	$\sim \sim J$✓		$1 \sim \vee D$
	4.	J		$3 \sim \sim D$
	5.	$J \supset Z$✓		$2 \ \&D$
	6.	$\sim Z$		$2 \ \&D$
	7.	$\sim J$	Z	$5 \supset D$
		×	×	

Since the truth-tree for the negation of the given sentence is closed, the given sentence is truth-functionally true.

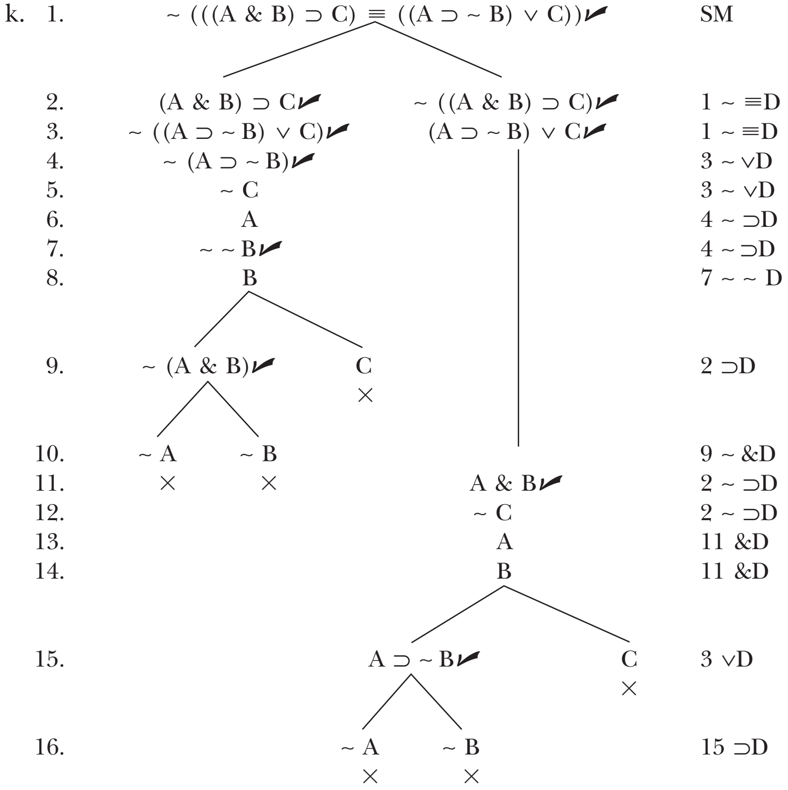


Since the truth-tree for the negation of the given sentence is closed, the given sentence is truth-functionally true.

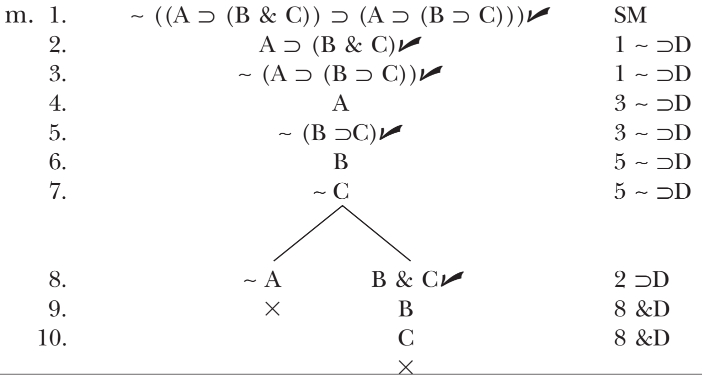


The tree for the negation of the sentence is not closed. Therefore the sentence is not truth-functionally true. The recoverable set of truth-value assignments is

A	B
T	**F**

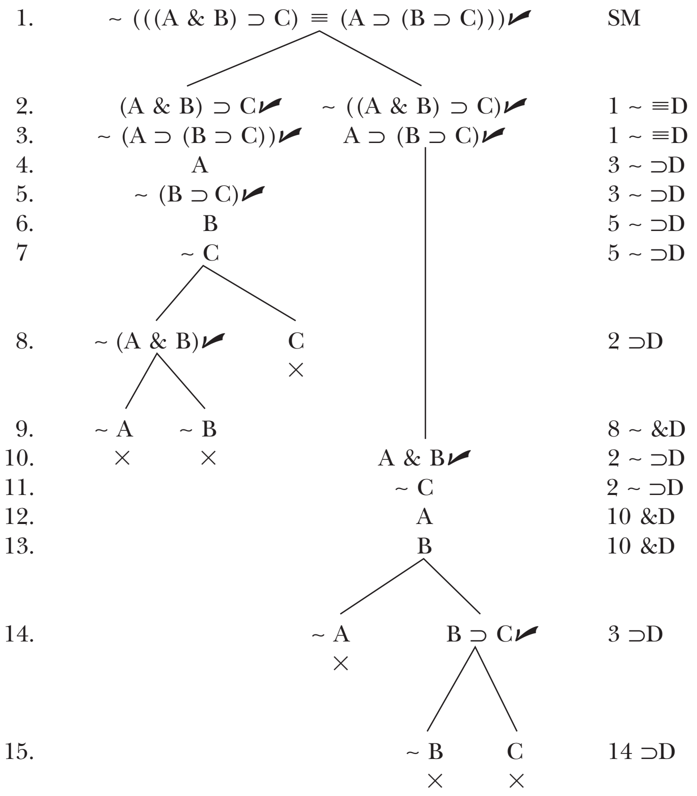


The tree for the negation of the sentence is closed. Therefore the sentence is truth-functionally true.



The tree for the negation of the sentence is closed. Therefore the sentence is truth-functionally true.

o.



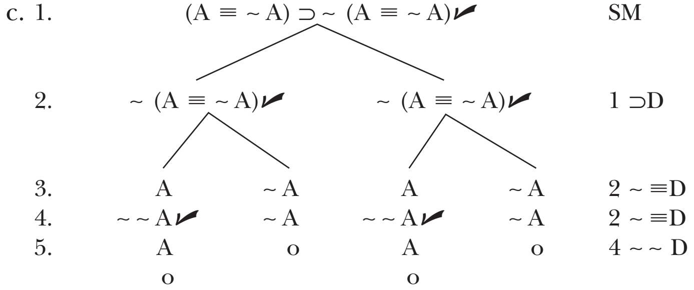
The tree for the negation of the sentence is closed. Therefore the sentence is truth-functionally true.

3.a. 1. ~ (~ A ⊃ A)✔ SM
2. ~ A 1 ~ ⊃D
3. ~ A 1 ~ ⊃D
o

The tree for the sentence does not close. Therefore the sentence is not truth-functionally false. The recoverable set of truth-value assignments is

A
F

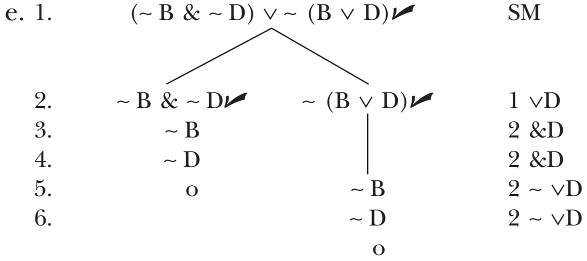
Since not all sets of truth-value assignments are recoverable, the sentence is not truth-functionally true. Therefore it is truth-functionally indeterminate.



The tree for the sentence does not close. Therefore the sentence is not truth-functionally false. The recoverable sets of truth-value assignments are

A
T
F

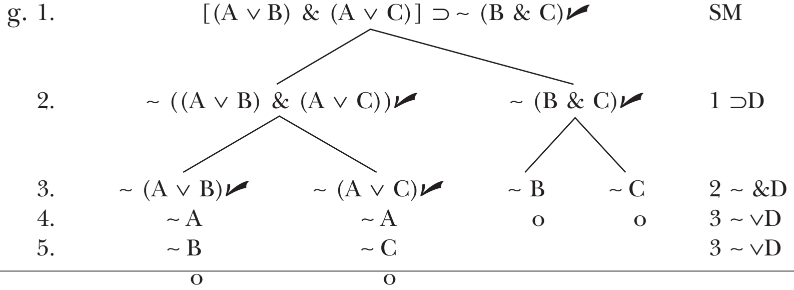
Since all sets of truth-value assignments are recoverable, the sentence is truth-functionally true.



The tree for the sentence does not close. Therefore the sentence is not truth-functionally false. The recoverable set of truth-value assignments is

B	D
F	**F**

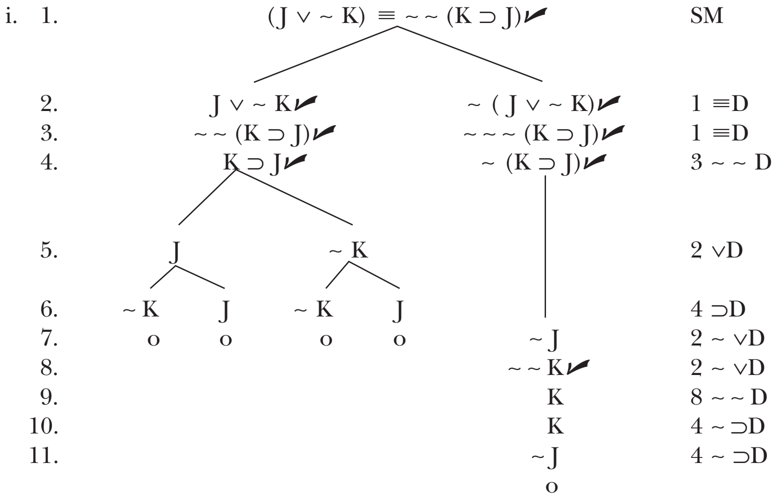
Since not all sets of truth-value assignments are recoverable, the sentence is not truth-functionally true. Therefore it is truth-functionally indeterminate.



The tree for the sentence does not close. Therefore the sentence is not truth-functionally false. The recoverable sets of truth-value assignments are

A	B	C
F	**F**	**T**
F	**F**	**F**
F	**T**	**F**
T	**F**	**T**
T	**F**	**F**
T	**T**	**F**

Since not all truth-value assignments are recoverable, the sentence is not truth-functionally true. Therefore it is truth-functionally indeterminate.

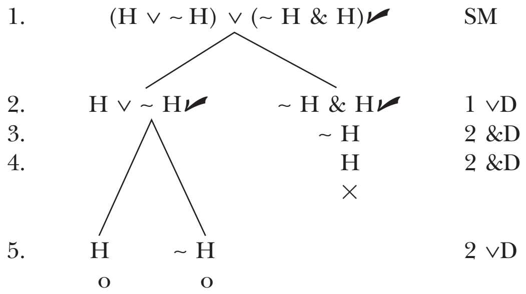


The tree for the sentence does not close. Therefore the sentence is not truth-functionally false. The recoverable sets of truth-value assignments are

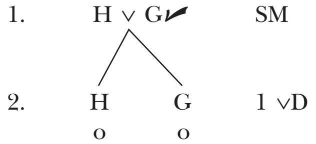
J	K
T	**F**
T	**T**
F	**T**
F	**F**

Since all truth-value assignments are recoverable, the sentence is truth-functionally true.

4.a. False. A tree for a truth-functionally true sentence can have some open and some closed branches. ‘(H ∨ ~ H) ∨ (~ H & H)’ is clearly truth-functionally true, inasmuch as its left disjunct is truth-functionally true. Yet the tree for this sentence has two open branches and one closed branch.



c. False. Many truth-functionally indeterminate sentences have completed trees all of whose branches are open. A simple example is



e. False. Some such unit sets open trees; for example, **P** ∨ **Q** does, but not all such unit sets have open trees. For example, **P** & **Q** has a closed tree if **P** is ‘H & G’ and **Q** is ‘~ H & K’.

1.	(H & G) & (~ H & K)✓	SM
2.	H & G✓	1 &D
3.	~ H & K✓	1 &D
4.	H	2 &D
5.	G	2 &D
6.	~ H	3 &D
7.	K	3 &D
	×	

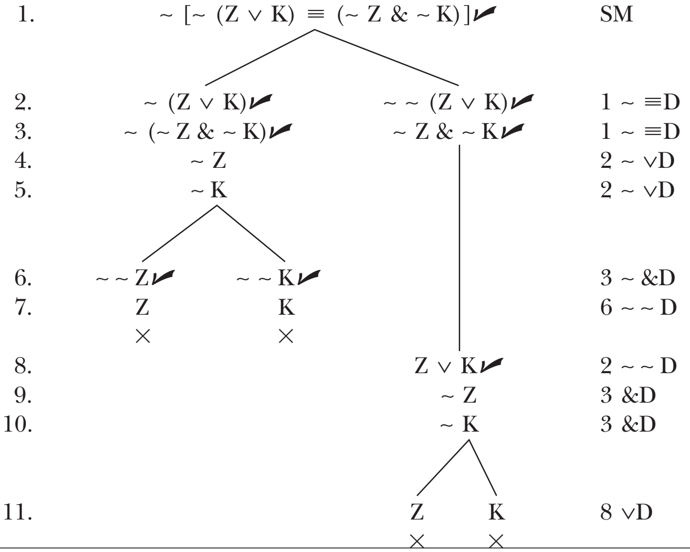
g. The claim is false. If **P** and **Q** are both truth-functionally true, then **P** & **Q**, **P** ∨ **Q**, **P** ⊃ **Q**, and **P** ≡ **Q** are also truth-functionally true. Therefore the unit set of each is truth-functionally consistent and will not have a closed truth-tree. But each may still have a tree with one or more closed branch. For example, if **P** is ‘(A ∨ ~ A) ∨ (B & ~ B)’ then **P** & **Q**, **P** ∨ **Q**, and **P** ≡ **Q** will each have at least one closed branch—the one resulting from the decomposition of ‘B & ~ B’. And if **P** is ‘A ∨ ~ A’ and **Q** is ‘B ∨ ~ B’, then the tree for **P** ⊃ **Q** will have a closed branch, the one resulting from the occurrence of ‘~ (A ∨ ~ A)’ on line 2 of the tree for this sentence.

i. The claim is false. Given that both **P** and **Q** are truth-functionally false, **P** & **Q** and **P** ∨ **Q** will also be truth-functionally false, and hence will have closed truth-trees. However, **P** ⊃ **Q** and **P** ≡ **Q** will both be truth-functionally true. (The only way **P** ⊃ **Q** could fail to be truth-functionally true would be there to be a truth-value assignment on which **P** is true and **Q** is false, but there is no truth-value assignment on which **P** is true since **P** is truth-functionally false. The only way **P** ≡ **Q** could fail to be truth-functionally true would be for there to be a truth-value assignment on which **P** and **Q** have different truth-values. But then there would have to be an assignment on which one of **P** and **Q** is true, but there can be no such assignment since both **P** and **Q** are truth-functionally false.) And sentences that are truth-functionally true have completed truth-trees that are open, not closed.

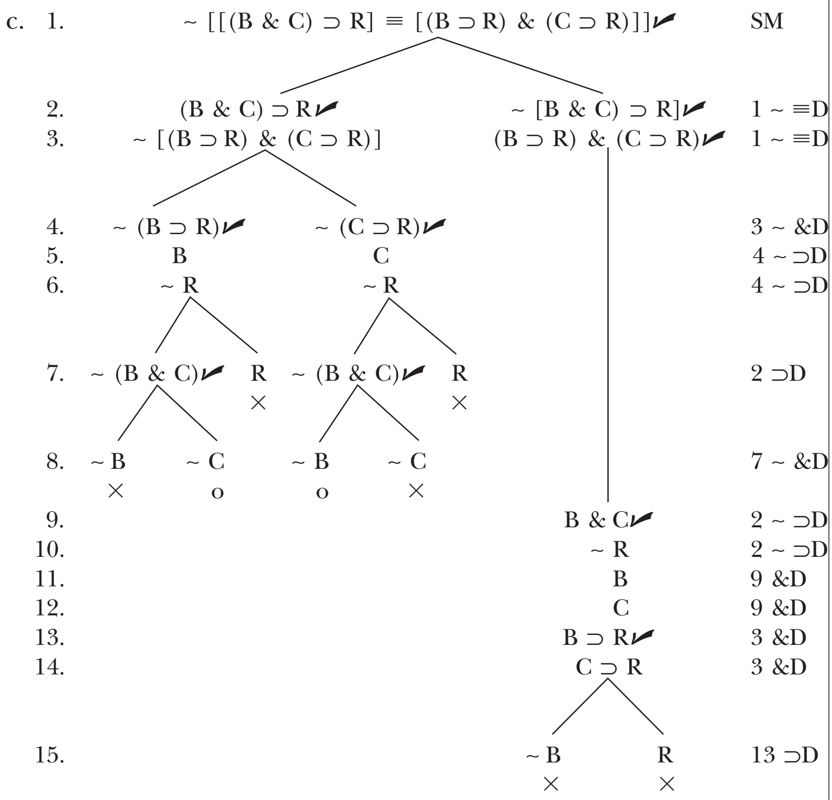
k. The claim is false. If **P** is, as stated, truth-functionally true and **Q** is truth-functionally false, then **P** & **Q**, **P** ⊃ **Q**, and **P** ≡ **Q** will all be truth-functionally false. **P** & **Q** so because there will be no truth-value assignment on which **P** and **Q** are both true (because **Q** is truth-functionally false. Hence **P** & **Q** will have a closed truth-tree (one on which every branch is closed). Similarly, **P** ⊃ **Q** will be false on every truth-value assignment because **P** will be true and **Q** false on every assignment. So the tree for **P** ⊃ **Q** will also be closed. **P** ≡ **Q** will be truth-functionally false because on every truth-value assignment **P** will be true and **Q** false, so there will be no assignment on which **P** and **Q** have the same truth-value, that is, no assignment on which **P** ≡ **Q** is true. So the tree for **P** ≡ **Q** will be closed. However, **P** ∨ **Q** will be truth-functionally true, because **P** is truth-functionally true. Line 2 of the tree will contain **P** on the left branch and **Q** on the right. Because **P** is truth-functionally true, subsequent work on the left branch will yield at least one (in fact at least two) completed open branches (see answer to exercise h). The right branch, that which has **Q** at the top, will become a closed branch because **Q** is truth-functionally false.

Section 4.6E

1.a.



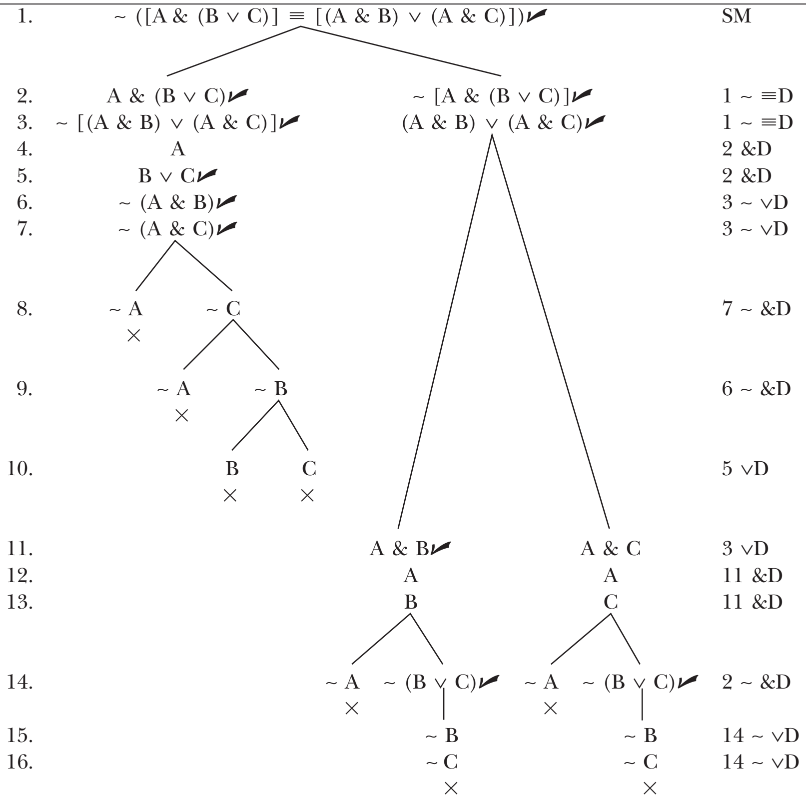
Our truth-tree for the negation of the biconditional of the sentences we are testing, '~ (Z ∨ K)' and '~ Z & ~ K', is closed. Therefore that negation is truth-functionally false, the biconditional it is a negation of is truth-functional true, and the sentences we are testing are truth-functionally equivalent.



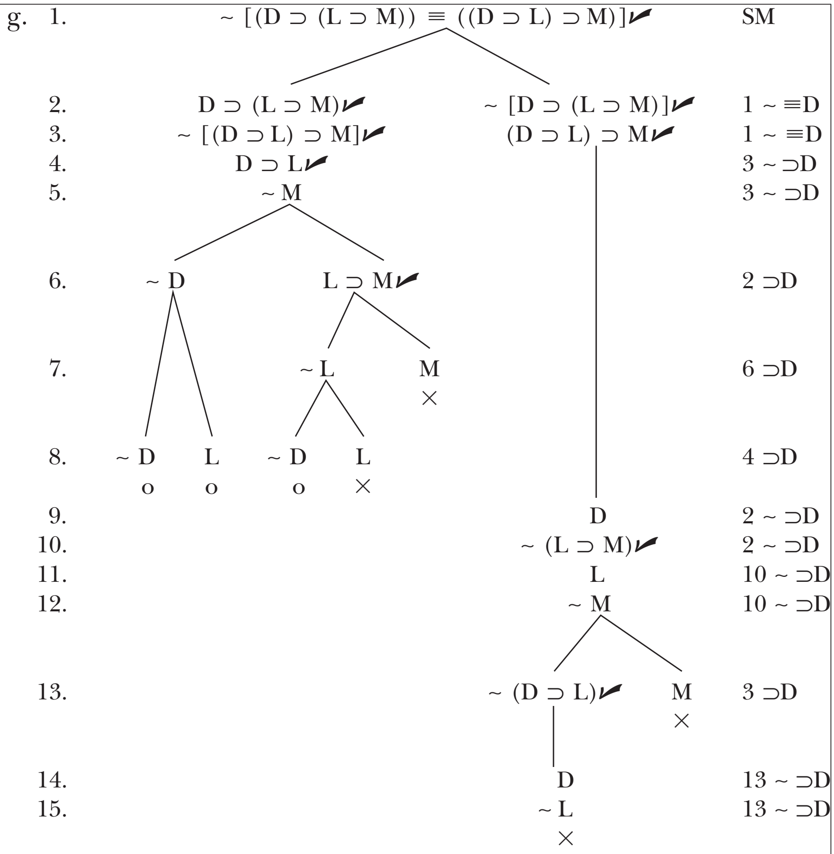
Since our truth-tree for the negation of the biconditional of the sentences we are testing is open, those sentences are not truth-functionally equivalent. The recoverable sets of truth-value assignments are

B	C	R
T	**F**	**F**
F	**T**	**F**

e.

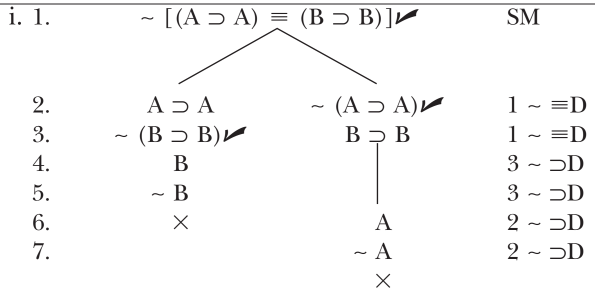


Since our truth-tree for the negation of the biconditional of the sentences we are testing is closed, those sentences are truth-functionally equivalent.

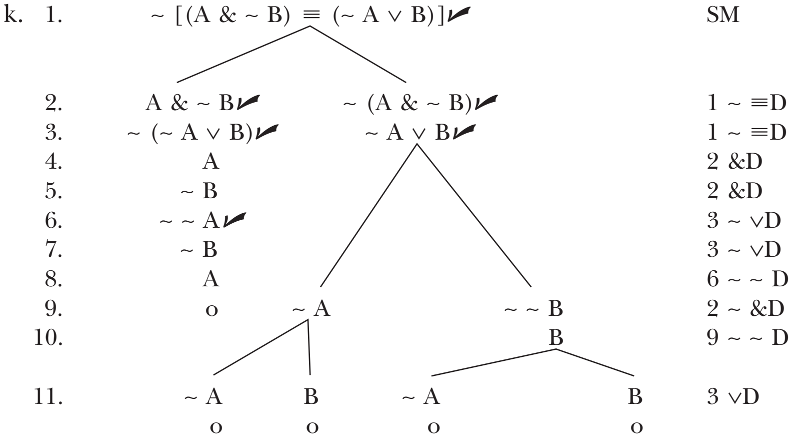


Since our truth-tree for the negation of the biconditional of the sentences we are testing is open, those sentences are not truth-functionally equivalent. The recoverable sets of truth-value assignments are

D	L	M
F	**T**	**F**
F	**F**	**F**

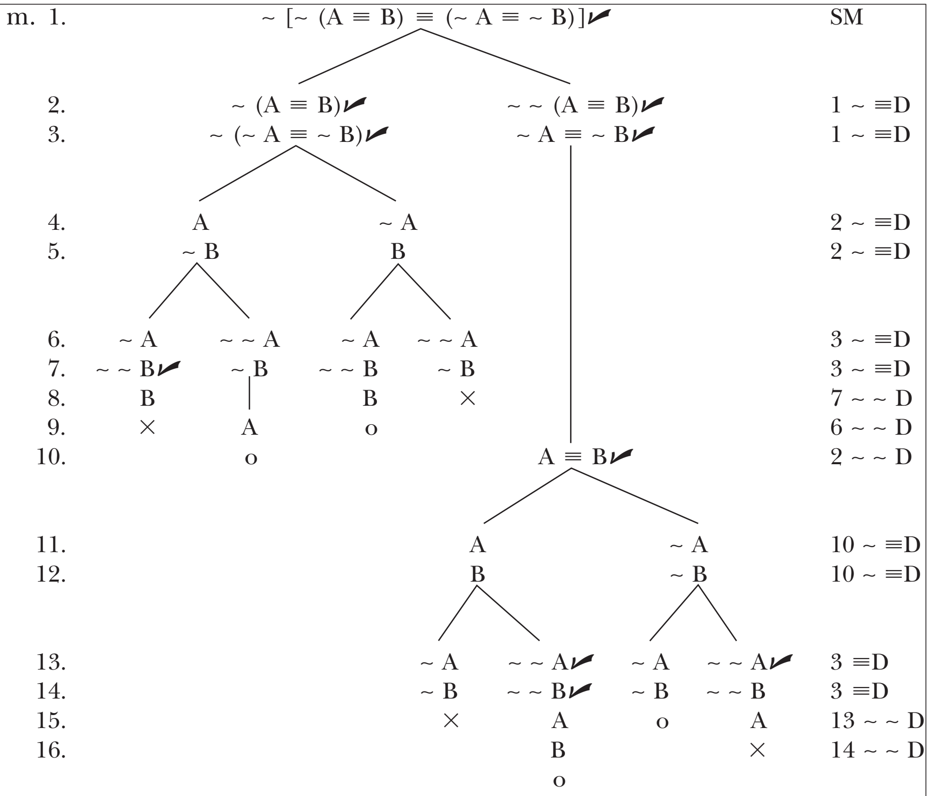


Since the truth-tree is closed, the sentences being tested are truth-functionally equivalent.



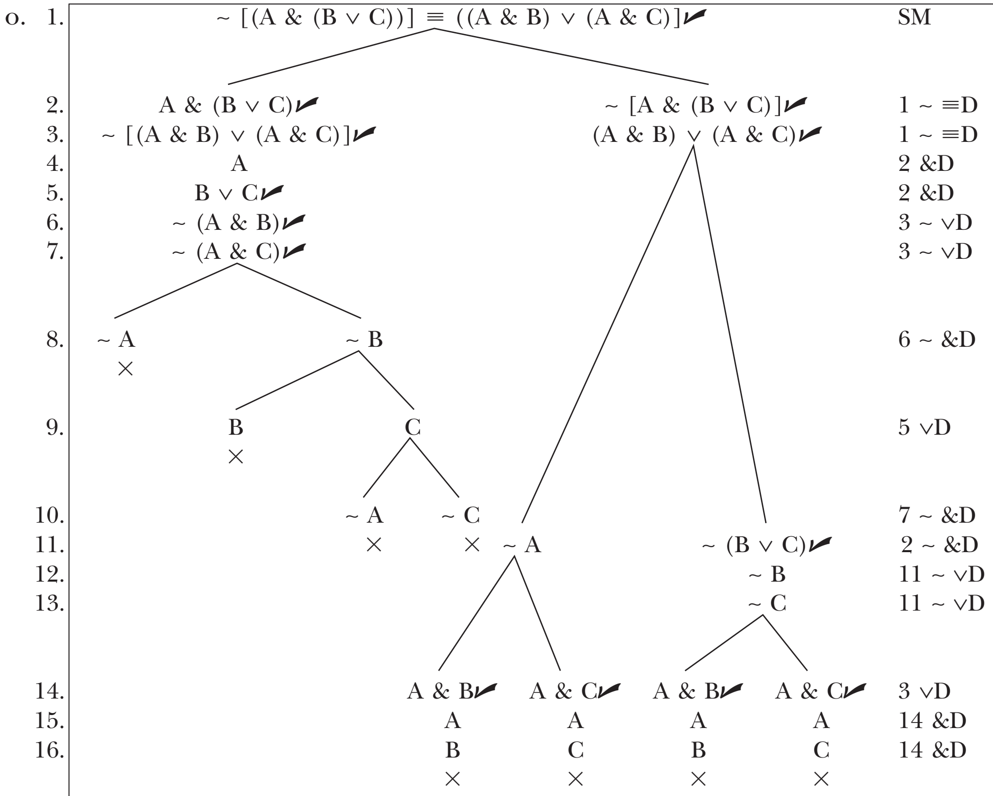
Since the truth-tree is not closed, the sentences being tested are not truth-functionally equivalent. The recoverable sets of truth-value assignments are

A	B
T	**F**
F	**F**
F	**T**
T	**T**



Since the truth-tree has at least one completed open branch, the sentences being tested are not truth-functionally equivalent. The recoverable sets of truth-value assignments are

A	B
T	**T**
F	**F**



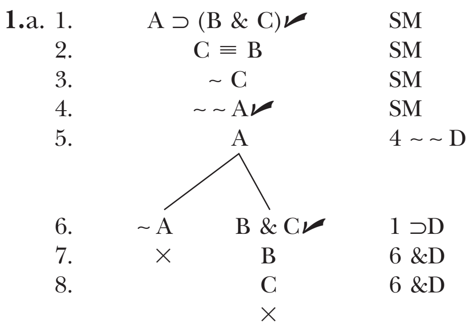
Since the truth-tree is closed, the sentences being tested are truth-functionally equivalent.

2.a. True. If **P** and **Q** are truth-functionally equivalent, their biconditional is truth-functionally true. And all truth-functionally true sentences have completed open trees.

c. False. The tree for the set {**P**, **Q**} may close, for **P** and **Q** may both be truth-functionally false. Remember that all truth-functionally false sentences are truth-functionally equivalent and a set composed of one or more truth-functionally false sentences has a closed tree.

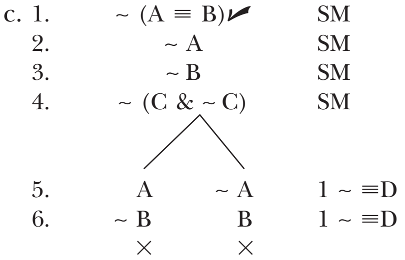
Section 4.7E

1.a.



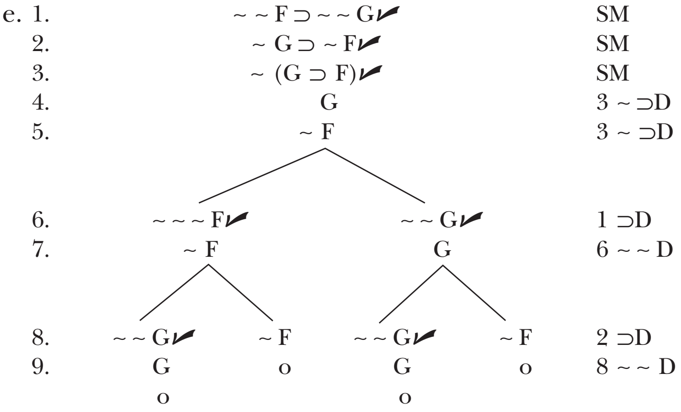
Our tree is closed, so the set {A ⊃ (B & C), C ≡ B, ~ C} does truth-functionally entail '~ A'.

c.



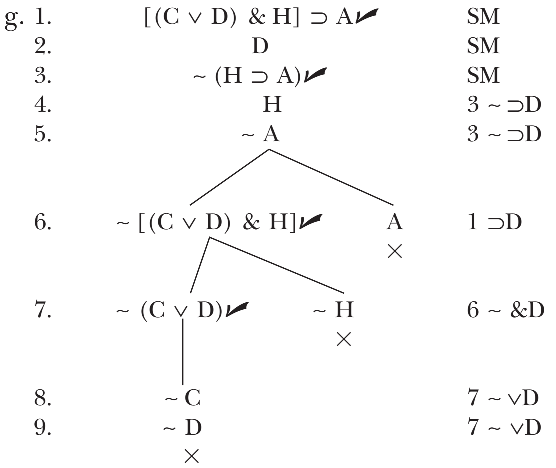
Our tree is closed, so the set {~ (A ≡ B), ~ A, ~ B} does truth-functionally entail 'C & ~ C'.

e.

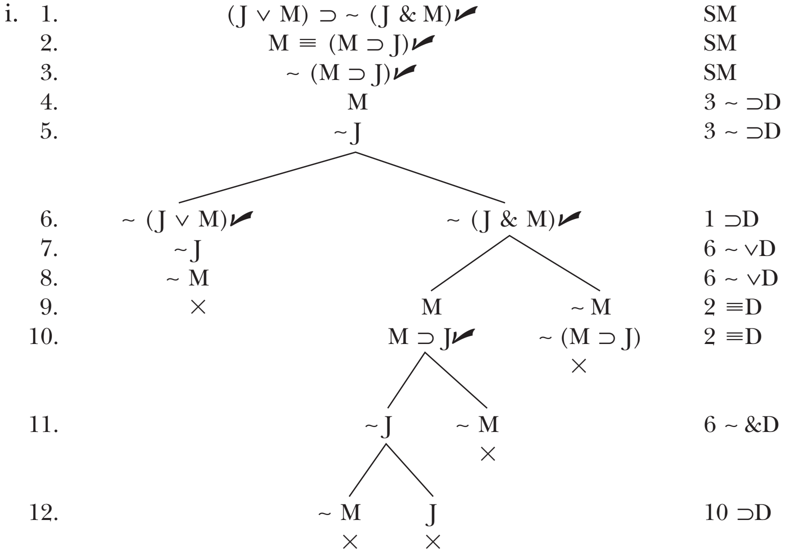


Our truth-tree is open, so the set {~ ~ F ⊃ ~ ~ G, ~ G ⊃ ~ F} does not truth-functionally entail ‘G ⊃ F’. The recoverable set of truth-value assignments is

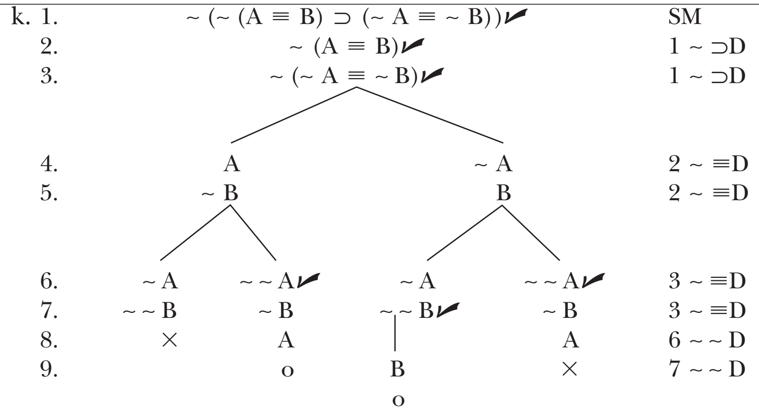
F	G
F	**T**



Our truth-tree is closed, so the given set does truth-functionally entail ‘H ⊃ A’.

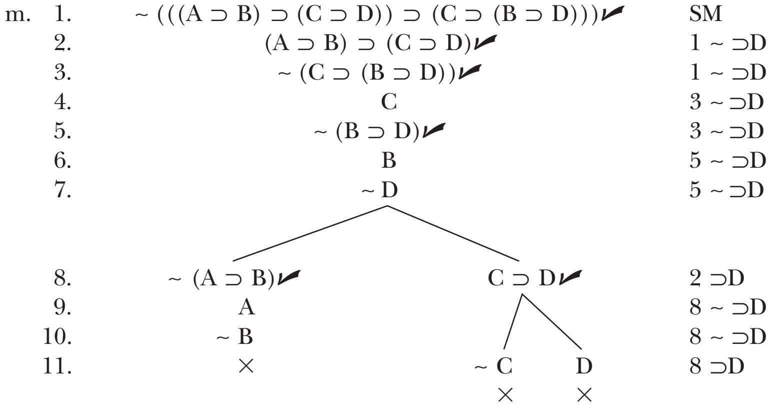


The tree is closed, so the set {(J ∨ M) ⊃ ~ (J & M), M ≡ (M ⊃ J)} does truth-functionally entail ‘M ⊃ J’.



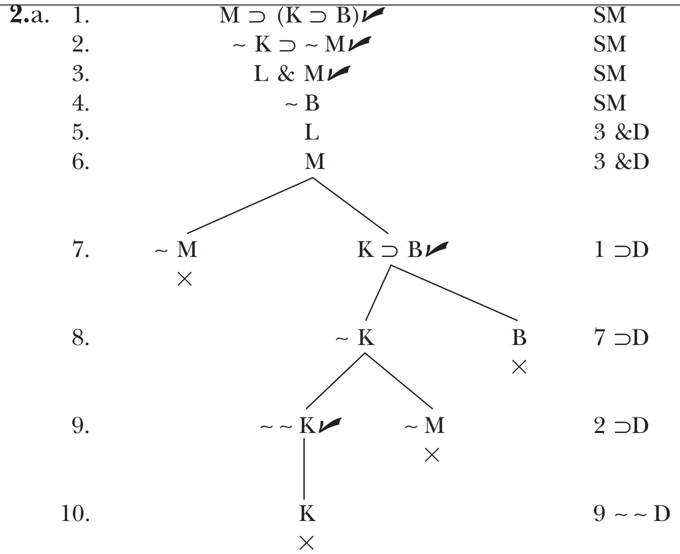
Our truth-tree is open, so the empty set does not truth-functionally entail '~ (A ≡ B) ⊃ (~ A ≡ ~ B)'. The recoverable sets of truth-value assignments are

A	B
T	**F**
F	**T**



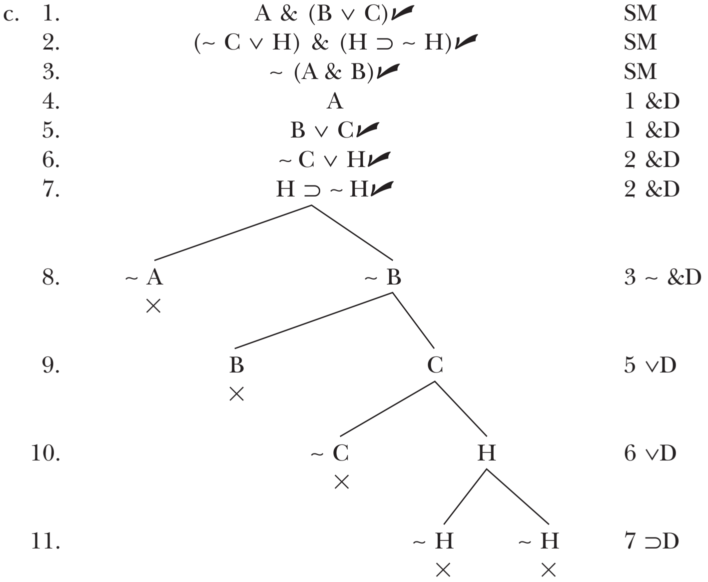
The tree is closed, so the empty set does truth-functionally entail '[(A ⊃ B) ⊃ (C ⊃ D)] ⊃ [C ⊃ (B ⊃ D)]'.

2.a.

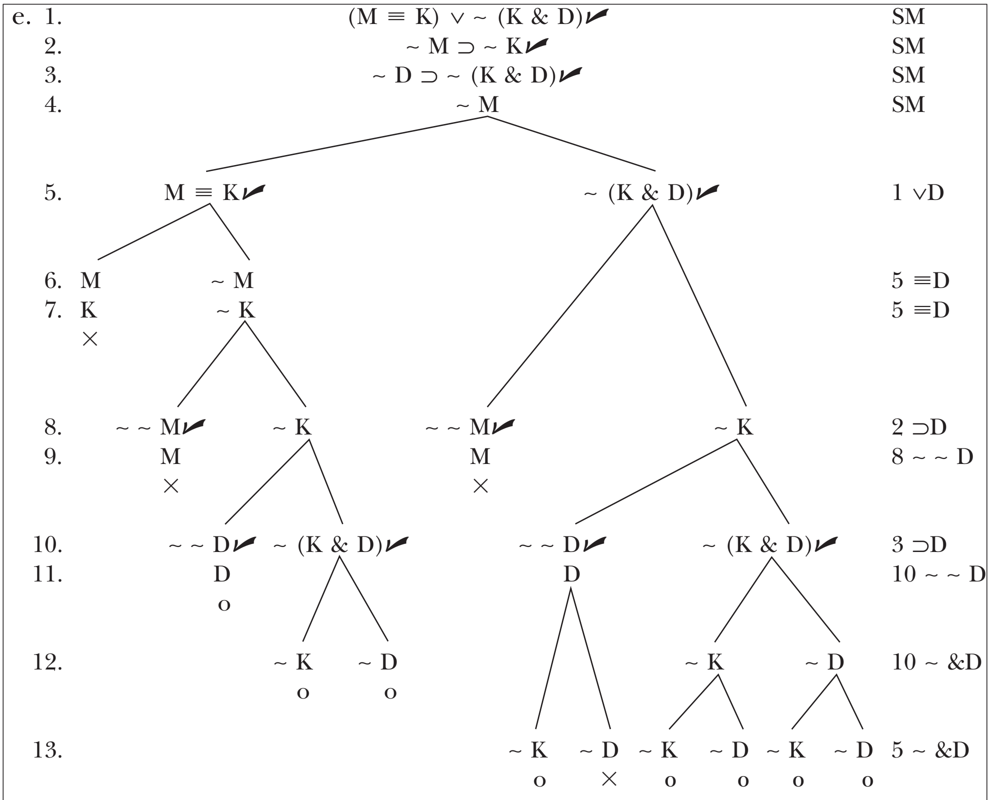


Our truth-tree for the premises and the negation of the conclusion of the argument we are testing is closed. Therefore there is no truth-value assignment on which the premises and the negation of the conclusion are all true, hence no assignment on which the premises are true and the conclusion false. So the argument is truth-functionally valid.

c.

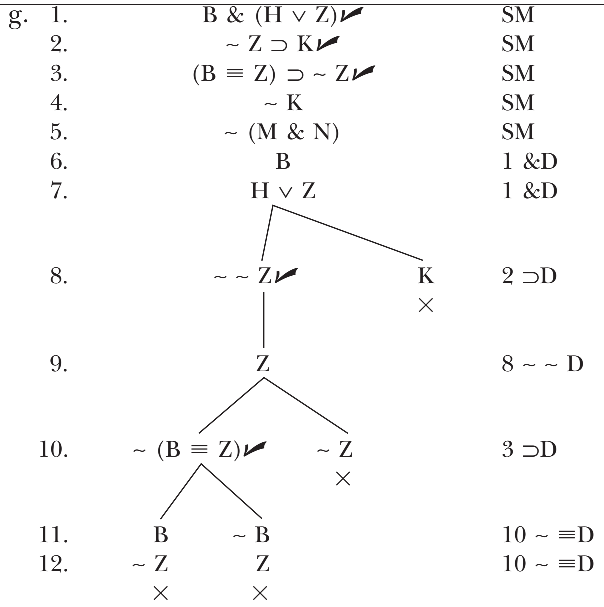


Our truth-tree for the premises and the negation of the conclusion of the argument we are testing is closed. Therefore the argument is truth-functionally valid.

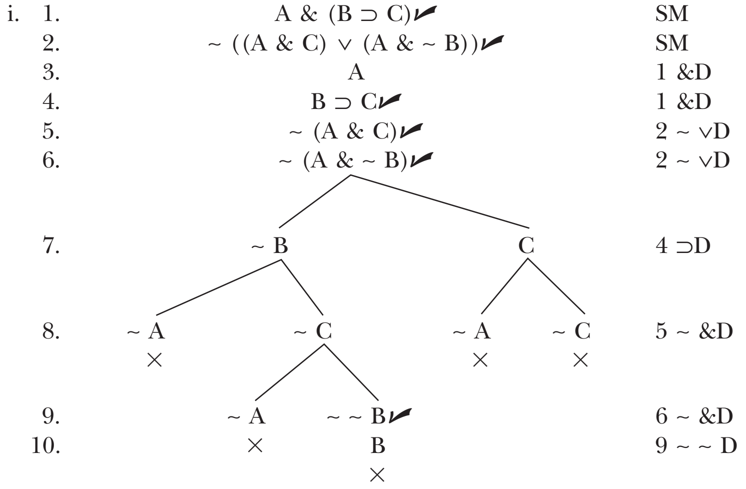


Our truth-tree for the premises and the negation of the conclusion of the argument we are testing is open. Therefore that argument is truth-functionally invalid. The recoverable sets of truth-value assignments are

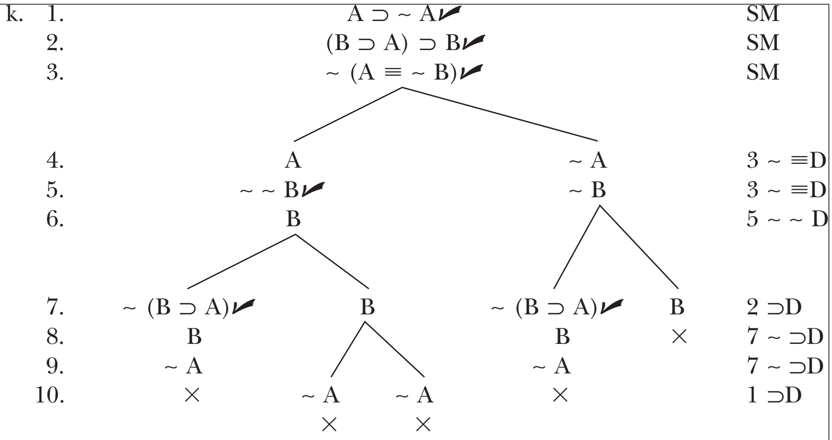
D	K	M
T	**F**	**F**
F	**F**	**F**



Our truth-tree for the premises and the negation of the conclusion of the argument we are testing is closed. Therefore that argument is truth-functionally valid. Notice that our tree closed before we decomposed the negation of the conclusion. Thus the premises of the argument form a truth-functionally inconsistent set, and therefore those premises and any conclusion constitute a truth-functionally valid argument, even where the conclusion has no atomic components in common with the premises.



Our truth-tree for the premise and the negation of the conclusion is closed. Therefore the argument is truth-functionally valid.

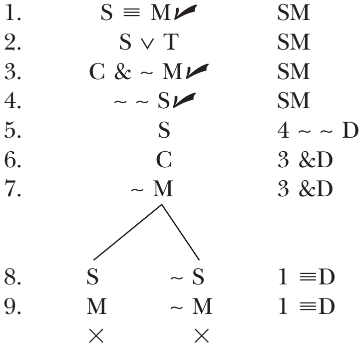


Our truth-tree for the premise and the negation of the conclusion is closed. Therefore the argument is truth-functionally valid.

3.a. In symbolizing the argument we use the following abbreviations:

C: Members of Congress claim to be sympathetic to senior citizens.
M: More money will be collected through social security taxes.
S: The social security system will succeed.
T: Many senior citizens will be in trouble.

Here is our tree for the premises and the negation of the conclusion:



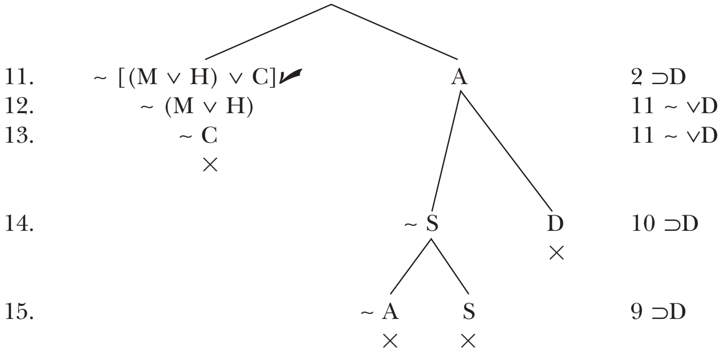
Since our truth-tree is closed, the argument is truth-functionally valid.

c. In symbolizing the argument we use the following abbreviations:

A: The President acts quickly.
C: The President is pressured by senior citizens.
D: Senior citizens will be delighted.
H: The President is pressured by members of the House.
M: The President is pressured by members of the Senate.
S: The social security system will be saved.

Here is our tree for the premises and the negation of the conclusion.

1.	$(A \supset S) \,\&\, (S \supset D)$ ✓	SM
2.	$[(M \vee H) \vee C] \supset A$ ✓	SM
3.	$\sim (M \vee H) \,\&\, C$ ✓	SM
4.	$\sim D$	SM
5.	$\sim (M \vee H)$ ✓	3 &D
6.	C	3 &D
7.	$\sim M$	5 ~ ∨D
8.	$\sim H$	5 ~ ∨D
9.	$A \supset S$ ✓	1 &D
10.	$S \supset D$ ✓	1 &D

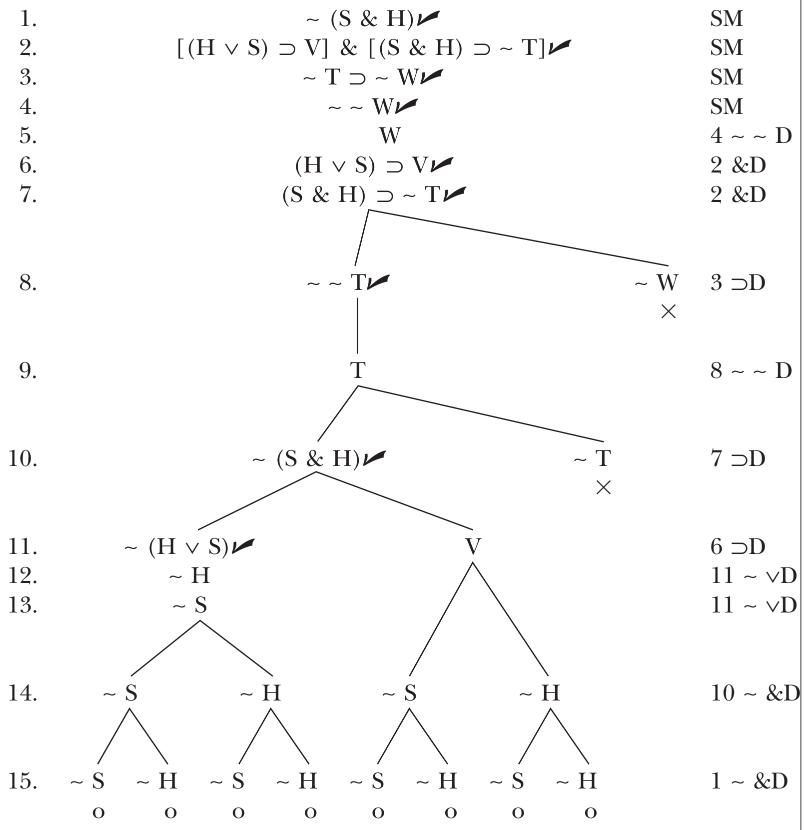


Since our tree is closed, the argument is truth-functionally valid.

e. In symbolizing the argument we use the following abbreviations:

H: The House of Representatives will pass the bill.
S: The Senate will pass the bill.
T: The President will be pleased.
V: The voters will be pleased.
W: All the members of the White House will be happy.

Here is our tree for the premises and the negation of the conclusion.

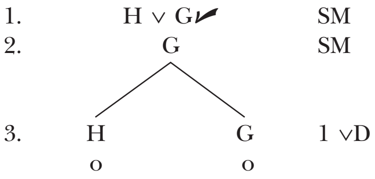


Since our truth-tree is open, the argument is truth-functionally invalid. The recoverable sets of truth-value assignments are

H	S	T	W	V
T	**F**	**T**	**T**	**T**
F	**T**	**T**	**T**	**T**
F	**F**	**T**	**T**	**T**
F	**F**	**T**	**T**	**F**

4.a. The first of the following arguments is truth-functionally invalid, the second truth-functionally valid. In each case the tree for the premise and the conclusion is open. This demonstrates that constructing a tree for the premises of an argument and the conclusion of the argument and finding that the tree has a completed open branch establishes neither that the argument is truth-functionally valid nor that it is truth-functionally invalid.

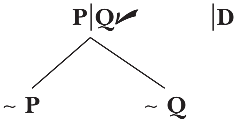
$$\frac{H \lor G}{G} \qquad\qquad \frac{H \,\&\, G}{G}$$



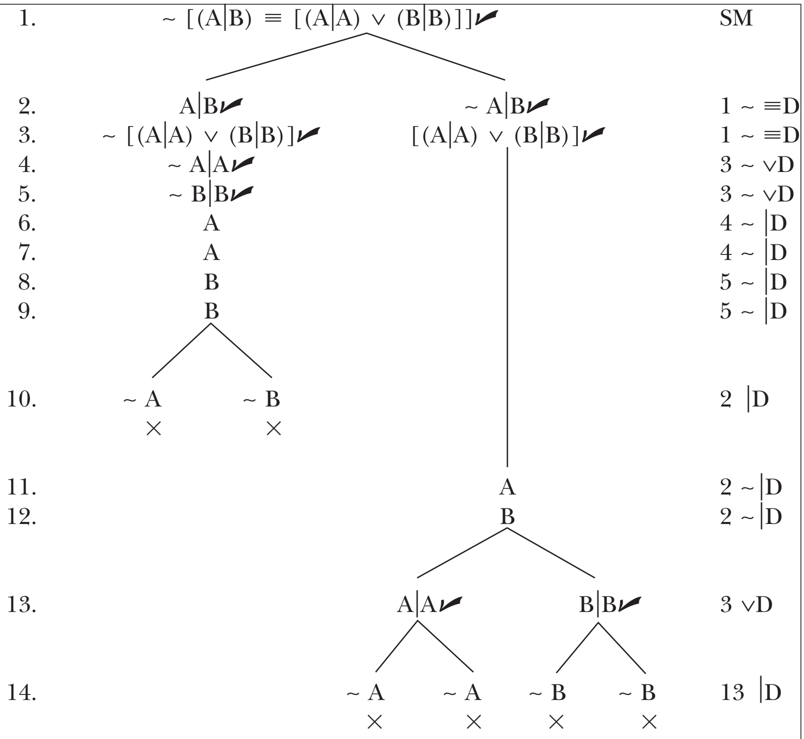
1.	H & G✓	SM
2.	G	SM
3.	H	1 &D
4.	G	1 &D
	o	

c. Since constructing a tree for the premises of an argument and the conclusion, whether the tree be open (see answer to a above) or closed (see answer to b above), establishes neither that the argument is truth-functionally valid nor that it is truth-functionally invalid, there is clearly no useful information to be gained by constructing such a tree.

5. The needed rules are



~ P|Q✓ **~|D**
P
Q



The truth-tree is closed. Therefore the sentences we are testing are truth-functionally equivalent.

CHAPTER FIVE

Exercises 5.1.1E

1.a. Derive: $A \& B$

1	A	Assumption
2	$A \supset B$	Assumption
3	B	1, 2 $\supset$E
4	$A \& B$	1, 3 &I

c. Derive: $A \supset (\sim C \& \sim B)$

1	$A \supset (\sim B \& \sim C)$	Assumption
2	A	A / $\supset$I
3	$\sim B \& \sim C$	1, 2 $\supset$E
4	$\sim B$	3 &E
5	$\sim C$	3 &E
6	$\sim C \& \sim B$	4, 5 &I
7	$A \supset (\sim C \& \sim B)$	1–6 $\supset$I

e. Derive: $\sim A \supset [B \& (D \& C)]$

1	$\sim A \supset B$	Assumption
2	$B \supset D$	Assumption
3	$\sim A \supset C$	Assumption
4	$\sim A$	A / $\supset$I
5	B	1, 4 $\supset$E
6	D	2, 5 $\supset$E
7	C	3, 6 $\supset$E
8	$D \& C$	6, 7 &I
9	$B \& (D \& C)$	5, 8 &I
10	$\sim A \supset [B \& (D \& C)]$	4–9 $\supset$I

g. Derive: $[(K \vee L) \supset I] \& [(K \vee L) \supset \sim J]$

1	$(K \vee L) \supset (I \& \sim J)$	Assumption
2	$K \vee L$	A / $\supset$I
3	$I \& \sim J$	1, 2 $\supset$E
4	I	3 &E
5	$(K \vee L) \supset I$	2–4 $\supset$I
6	$K \vee L$	A / $\supset$I
7	$I \& \sim J$	1, 6 $\supset$E
8	$\sim J$	7 &E
9	$(K \vee L) \supset \sim J$	6–8 $\supset$E
10	$[(K \vee L) \supset I] \& [(K \vee L) \supset \sim J]$	5, 9 &I

i. Derive: A ⊃ (B ⊃ C)

1	(A & B) ⊃ C	Assumption
2	A	A / ⊃I
3	B	A / ⊃I
4	A & B	2, 3 & I
5	C	1, 4 ⊃E
6	B ⊃ C	3–5 ⊃I
7	A ⊃ (B ⊃ C)	2–6 ⊃I

k. Derive: (A & B) ⊃ (C & D)

1	(B & A) ⊃ (D & C)	Assumption
2	A & B	A / ⊃I
3	B	2 &E
4	A	2 &E
5	B & A	3, 4 &I
6	D & C	1, 5 ⊃E
7	C	6 &E
8	D	6 &E
9	C & D	7, 8 &I
10	(A & B) ⊃ (C & D)	2–9 ⊃I

m. Derive: (A & B) ⊃ E

1	A ⊃ C	Assumption
2	B ⊃ D	Assumption
3	(C & D) ⊃ E	Assumption
4	A & B	A / ⊃I
5	A	4 &E
6	B	4 &E
7	C	1, 5 ⊃E
8	D	2, 6 ⊃E
9	C & D	7, 8 &I
10	E	3, 9 ⊃E
11	(A & B) ⊃ E	4–10 ⊃1

Exercises 5.1.2E

1.a. Derive: ~ G

1	(G ⊃ I) & ~ I	Assumption
2	G	A /~ I
3	G ⊃ I	1 &E
4	I	2, 3 ⊃E
5	~ I	1 &E
6	~ G	2–5 ~ I

c. Derive: $\sim \sim B$

1	$\sim B \supset A$	Assumption
2	$\sim B \supset \sim A$	Assumption
3	$\quad \sim B$	A /~ I
4	$\quad A$	1, 3 ⊃E
5	$\quad \sim A$	2, 3 ⊃E
6	$\sim \sim B$	3–5 ~ I

e. Derive: A

1	$(\sim A \supset \sim B) \& (\sim B \supset B)$	Assumption
2	$\quad \sim A$	A /~ E
3	$\quad \sim A \supset \sim B$	1 &E
4	$\quad \sim B$	2, 3 ⊃E
5	$\quad \sim B \supset B$	1 &E
6	$\quad B$	4, 5 ⊃E
7	A	2–6 ~ E

Exercises 5.1.3E

1.a. Derive: $B \vee (K \vee G)$

1	K	Assumption
2	$K \vee G$	I ∨I
3	$B \vee (K \vee G)$	2 ∨I

c. Derive: $D \vee E$

1	$E \vee D$	Assumption
2	$\quad E$	A / ∨E
3	$\quad D \vee E$	2 ∨I
4	$\quad D$	A / ∨E
5	$\quad D \vee E$	4 ∨ I
6	$D \vee E$	1, 2–3, 4–5 ∨E

e. Derive: F

1	$\sim E \vee F$	Assumption
2	$\sim E \supset F$	Assumption
3	$\quad \sim E$	A / ∨E
4	$\quad F$	2, 3 ⊃ E
5	$\quad F$	A / ∨E
6	$\quad F$	5 R
7	F	1, 3–4, 5–6 ∨E

Exercises 5.1.4E

1. a. Derive: L

1	K ≡(~ E & L)	Assumption
2	K	Assumption
3	~ E & L	1, 2 ≡E
4	L	3 &E

c. Derive: S & ~ A

1	(S ≡ ~ I) & N	Assumption
2	(N ≡ ~ I) & ~ A	Assumption
3	~ A	2 &E
4	N ≡ ~ I	2 &E
5	N	1 &E
6	~ I	4, 5 ≡E
7	S ≡ ~ I	1 &E
8	S	6, 7 ≡E
9	S & ~ A	3, 8 &I

e. Derive: E ≡ O

1	(E ⊃ T) & (T ⊃ O)	Assumption
2	O ⊃ E	Assumption
3	E	A / ≡I
4	E ⊃ T	1 &E
5	T	3, 4 ⊃E
6	T ⊃ O	1 &E
7	O	5, 6 ⊃E
8	O	A / ≡I
9	E	2, 8 ⊃E
10	E ≡ O	3–7, 8–9 ≡I

Exercises 5.3E

1. Derivability

a. Derive: A ⊃ (A & B)

1	A ⊃ B	Assumption
G	A ⊃ (A & B)	2–__ ⊃I

Derive: $A \supset (A \,\&\, B)$

1	$A \supset B$	Assumption
2	A	A / ⊃I
3	B	1, 2 ⊃E
4	$A \,\&\, B$	2, 3 &I
5	$A \supset (A \,\&\, B)$	2–5 ⊃I

c. Derive: $L \equiv K$

1	$(K \supset L) \,\&\, (L \supset K)$	Assumption
2	L	A / ≡I
G	K	
	K	A / ≡I
G	L	
G	$L \equiv K$	2–__, __ –__ ≡I

Derive: $L \equiv K$

1	$(K \supset L) \,\&\, (L \supset K)$	Assumption
2	L	A / ≡I
3	$L \supset K$	1 &E
4	K	2, 3 ⊃E
5	K	A / ≡I
6	$K \supset L$	1 &E
7	L	5, 6 ⊃ E
8	$L \equiv K$	2–4, 5–7 ≡I

e. Derive: C

1	$B \,\&\, {\sim} B$	Assumption
2	${\sim} C$	A /~ E
G	C	2–__ ~ E

Derive: C

```
1  | B & ~ B                     Assumption
   |---------
2  |  | ~ C                      A /~ E
   |  |------
3  |  | B                        1 &E
4  |  | ~ B                      1 &E
5  | C                           2–4 ~ E
```

g. Derive: D ⊃ B

```
 1 | A ⊃ C                       Assumption
 2 | (~ A ∨ C) ⊃ (D ⊃ B)         Assumption
   |--------------------
   |
   |
   |
   |
 G | ~ A ∨ C
 G | D ⊃ B                       2, __ ⊃E
```

Derive: D ⊃ B

```
 1 | A ⊃ C                       Assumption
 2 | (~ A ∨ C) ⊃ (D ⊃ B)         Assumption
   |--------------------
 3 |  | ~ (~ A ∨ C)              A /~ E
   |  |-----------------
 4 |  |  | A                     A /~ I
   |  |  |--------------
 5 |  |  | C                     1, 4 ⊃E
 6 |  |  | ~ A ∨ C               5 ∨I
 7 |  |  | ~ (~ A ∨ C)           3 R
 8 |  | ~ A                      4–7 ~I
 9 |  | ~ A ∨ C                  8 ∨I
10 |  | ~ (~ A ∨ C)              3 R
11 | ~ A ∨ C                     3, 10 ~ E
12 | D ⊃ B                       2, 11 ⊃E
```

i. Derive: B

1	A ⊃ B	Assumption
2	~ (B & ~ C) ⊃ A	Assumption
3	~ B	A /~ E
G	B	3-___ ~ E

Derive: B

1	A ⊃ B	Assumption
2	~ (B & ~ C) ⊃ A	Assumption
3	~ B	A /~ E
4	B & ~ C	A /~ I
5	B	4 &E
6	~ B	3 R
7	~ (B & ~ C)	4–6 ~ I
8	A	2, 7 ⊃E
9	B	1, 8 ⊃E
10	~ B	3 R
11	B	3–10 ~ E

k. Derive: B ∨ ~ C

1	A ∨ (B & C)	Assumption
2	C ⊃ ~ A	Assumption
3	A	A / ∨E
G	B ∨ ~ C	
	B & C	A / ∨E
G	B ∨ ~ C	
G	B ∨ ~ C	1, 3-___, ___-___ ∨E

Derive: B ∨ ~ C

```
 1 | A ∨ (B & C)                 Assumption
 2 | C ⊃ ~ A                     Assumption
   |------------
 3 |  | A                        A / ∨E
   |  |--------
 4 |  |  | C                     A /~ I
   |  |  |-----
 5 |  |  | ~ A                   2, 4 ⊃E
 6 |  |  | A                     3 R
 7 |  | ~ C                      4–6 ~ 1
 8 |  | B ∨ ~ C                  7 ∨I
   |
 9 |  | B & C                    A / ∨E
   |  |--------
10 |  | B                        9 &E
11 |  | B ∨ ~ C                  10 ∨I
12 | B ∨ ~ C                     1, 3–8, 9–11 ∨E
```

m. Derive: D ⊃ (F ⊃ C)

```
 1 | (A ∨ B) ⊃ C
 2 | (D ∨ E) ⊃ [(F ∨ G) ⊃ A]
   |----------------------------
 3 |  | D                        A / ⊃I
   |  |-------------------------
   |  |
   |  |
   |  |
   |  |
 G |  | F ⊃ C
 G | D ⊃ (F ⊃ C)                 3-__ ⊃I
```

Derive: D ⊃ (F ⊃ C)

```
 1 | (A ∨ B) ⊃ C
 2 | (D ∨ E) ⊃ [(F ∨ G) ⊃ A]
   |----------------------------
 3 |  | D                        A / ⊃I
   |  |-------------------------
 4 |  |  | F                     A / ⊃I
   |  |  |----------------------
 5 |  |  | D ∨ E                 3 ∨ I
 6 |  |  | (F ∨ G) ⊃ A           2, 5 ⊃E
 7 |  |  | F ∨ G                 4 ∨I
 8 |  |  | A                     6, 7 ⊃E
 9 |  |  | A ∨ B                 8 ∨I
10 |  |  | C                     1, 9 ⊃E
11 |  | F ⊃ C                    4–10 ⊃I
12 | D ⊃ (F ⊃ C)                 3–11 ⊃I
```

o. Derive: B ⊃ F

```
 1 │ A ⊃ ~ (B ∨ C)               Assumption
 2 │ (C ∨ D) ⊃ A                 Assumption
 3 │ ~ F ⊃ (D & ~ E)             Assumption
   ├──────────────
 4 │ │ B                         A / ⊃I
   │ ├──────────
   │ │
   │ │
 G │ │ F
 G │ B ⊃ F                       4-___ ⊃I
```

Derive: B ⊃ F

```
 1 │ A ⊃ ~ (B ∨ C)               Assumption
 2 │ (C ∨ D) ⊃ A                 Assumption
 3 │ ~ F ⊃ (D & ~ E)             Assumption
   ├──────────────
 4 │ │ B                         A / ⊃I
   │ ├──────────
 5 │ │ │ ~ F                     A / ~E
   │ │ ├──────
 6 │ │ │ D & ~ E                 3, 5 ⊃ E
 7 │ │ │ D                       6 &E
 8 │ │ │ C ∨ D                   7 ∨I
 9 │ │ │ A                       2, 8 ⊃ E
10 │ │ │ ~ (B ∨ C)               1, 9 ⊃E
11 │ │ │ B ∨ C                   4 ∨I
12 │ │ F                         5–11 ~ I
13 │ B ⊃ F                       4–12 ⊃I
```

q. Derive: H

```
 1 │ F ⊃ (G ∨ H)                 Assumption
 2 │ ~ (~ F ∨ H)                 Assumption
 3 │ ~ G                         Assumption
   ├──────────
 4 │ │ ~ H                       A /~ E
   │ ├───────
   │ │
   │ │
 G │ │ ~ F ∨ H
 G │ │ ~ (~ F ∨ H)               2 R
 G │ H                           4-___ ~E
```

q. Derive: H

1	│ F ⊃ (G ∨ H)	Assumption
2	│ ~ (~ F ∨ H)	Assumption
3	│ ~ G	Assumption
4	│ │ ~ H	A /~ E
5	│ │ │ F	A / ~ I
6	│ │ │ G ∨ H	1, 5 ⊃E
7	│ │ │ │ G	A / ∨E
8	│ │ │ │ │ ~ H	A / ~E
9	│ │ │ │ │ G	7 R
10	│ │ │ │ │ ~ G	3 R
12	│ │ │ │ H	8–10 ~ E
13	│ │ │ │ H	A / ∨E
14	│ │ │ │ ~ H	4 R
15	│ │ │ │ H	13 R
16	│ │ │ H	6, 7–12, 13–15 ∨E
17	│ │ │ ~ H	4 R
18	│ │ ~ F	5–17 ~E
19	│ │ ~ F ∨ H	18 ∨I
20	│ │ ~ (~ F ∨ H)	2 R
21	│ H	4–20 ~ E

2. Validity

a. Derive: A ⊃ C

1	│ A ⊃ ~ B	Assumption
2	│ ~ B ⊃ C	Assumption
3	│ │ A	A / ⊃I
4	│ │ ~ B	1, 3 ⊃E
5	│ │ C	2, 4 ⊃E
6	│ A ⊃ C	3–5 ⊃I

c. Derive: ~ B

1	│ A ≡ B	Assumption
2	│ ~ A	Assumption
3	│ │ B	A / ~ 1
4	│ │ A	1, 3 ≡E
5	│ │ ~ A	2 R
6	│ ~B	3–5 ~ 1

e. Derive: $A \supset [B \supset (C \supset D)]$

1	│ D	Assumption
2	│ │ A	A / ⊃I
3	│ │ │ B	A / ⊃I
4	│ │ │ │ C	A / ⊃I
5	│ │ │ │ D	1 R
6	│ │ │ $C \supset D$	4–5 ⊃I
7	│ │ $B \supset (C \supset D)$	3–6 ⊃I
8	│ $A \supset [B \supset (C \supset D)]$	2–7 ⊃I

g. Derive: $A \supset (D \supset C)$

1	│ $A \supset (B \supset C)$	Assumption
2	│ $D \supset B$	Assumption
3	│ │ A	A / ⊃I
4	│ │ │ D	A / ⊃I
5	│ │ │ $B \supset C$	1, 3 ⊃E
6	│ │ │ B	2, 4 ⊃E
7	│ │ │ C	5, 6 ⊃ E
8	│ │ $D \supset C$	4–7 ⊃I
9	│ $A \supset (D \supset C)$	3–8 ⊃I

i. Derive: $A \supset C$

1	│ $\sim A \vee B$	Assumption
2	│ $B \supset C$	Assumption
3	│ │ A	A / ⊃I
4	│ │ │ $\sim A$	A / ∨E
5	│ │ │ │ $\sim C$	A / ~ E
6	│ │ │ │ A	3 R
7	│ │ │ │ $\sim A$	4 R
8	│ │ │ C	5–7 ~ E
9	│ │ │ B	A / ∨E
10	│ │ │ C	2, 9 ⊃E
11	│ │ C	1, 4–8, 9–10 ∨E
12	│ $A \supset C$	3–11 ⊃ I

k. Derive: B

1	│ A ⊃ (C ⊃ B)	Assumption
2	│ ~ C ⊃ ~ A	Assumption
3	│ A	Assumption
4	│ │ ~ B	A / ~ E
5	│ │ C ⊃ B	1, 3 ⊃E
6	│ │ │ C	A / ~ I
7	│ │ │ B	5, 6 ⊃ E
8	│ │ │ ~ B	4 R
9	│ │ ~ C	6–8 ~ I
10	│ │ ~ A	2, 9 ⊃E
11	│ │ A	3 R
12	│ B	4–11 ~ E

*m. Derive: F & G

1	│ F ≡ G	Assumption
2	│ F ∨ G	Assumption
3	│ │ F	A / ∨E
4	│ │ F	3 R
5	│ │ G	A / ∨E
6	│ │ F	1, 5 ≡E
7	│ F	2, 3–4, 5–6 ∨E
8	│ G	1, 7 ≡E
9	│ F & G	7, 8 &I

3. Theorems

a. Derive: A ⊃ (A ∨ B)

1	│ │ A	A / ⊃I
2	│ │ A ∨ B	1 ∨I
3	│ A ⊃ (A ∨ B)	1–2 ⊃I

c. Derive: A ⊃ [B ⊃ (A & B)]

1	│ │ A	A / ⊃I
2	│ │ │ B	A / ⊃I
3	│ │ │ A & B	1, 2 &I
4	│ │ B ⊃ (A & B)	2–3 ⊃I
5	│ A ⊃ [B ⊃ (A & B)]	1–4 ⊃I

e. Derive: (A ≡ B) ⊃ (A ⊃ B)

```
1 │ │ A ≡ B                         A / ⊃I
  │ ├──────────────
2 │ │ │ A                           A / ⊃I
  │ │ ├────────────
3 │ │ │ B                           1, 2 ≡E
4 │ │ A ⊃ B                         2–3 ⊃I
5 │ (A ≡ B) ⊃ (A ⊃ B)               1–4 ⊃I
```

g. Derive: (A ⊃ B) ⊃ [(C ⊃ A) ⊃ (C ⊃ B)]

```
1 │ │ A ⊃ B                         A / ⊃I
  │ ├──────────────────────
2 │ │ │ C ⊃ A                       A / ⊃I
  │ │ ├────────────────────
3 │ │ │ │ C                         A / ⊃I
  │ │ │ ├──────────────────
4 │ │ │ │ A                         2, 3 ⊃E
5 │ │ │ │ B                         1, 4 ⊃E
6 │ │ │ C ⊃ B                       3–5 ⊃I
7 │ │ (C ⊃ A) ⊃ (C ⊃ B)             2–6 ⊃I
8 │ (A ⊃ B) ⊃ [(C ⊃ A) ⊃ (C ⊃ B)]   1–7 ⊃I
```

i. Derive: [(A ⊃ B) & ~ B] ⊃ ~ A

```
1 │ │ (A ⊃ B) & ~ B                 A / ⊃I
  │ ├──────────────────
2 │ │ │ A                           A / ⊃I
  │ │ ├────────────────
3 │ │ │ A ⊃ B                       1 &E
4 │ │ │ B                           2, 3 ⊃I
5 │ │ │ ~ B                         1 &E
6 │ │ ~ A                           2–5 ~ I
7 │ [(A ⊃ B) & ~ B] ⊃ ~ A           1–6 ⊃I
```

k. Derive: A ⊃ [B ⊃ (A ⊃ B)]

```
1 │ │ A                             A / ⊃I
  │ ├──────────────
2 │ │ │ B                           A / ⊃I
  │ │ ├────────────
3 │ │ │ │ A                         A / ⊃I
  │ │ │ ├──────────
4 │ │ │ │ B                         2 R
5 │ │ │ A ⊃ B                       3–4 ⊃I
6 │ │ B ⊃ (A ⊃ B)                   2–5 ⊃I
7 │ A ⊃ [B ⊃ (A ⊃ B)]               1–6 ⊃I
```

m. Derive: (A ⊃ B) ⊃ [~ B ⊃ ~ (A & D)]

1	A ⊃ B	A / ⊃I
2	~ B	A / ⊃I
3	A & D	A / ~ I
4	A	3 &E
5	B	1, 4 ⊃E
6	~ B	2 R
7	~ (A & D)	3–6 ~ I
8	~ B ⊃ ~ (A & D)	2–7 ⊃I
9	(A ⊃ B) ⊃ [~ B ⊃ ~ (A & D)]	1–8 ⊃I

4. Equivalence

a. Derive: A & ~ A

1	B & ~ B	Assumption
2	~ (A & ~ A)	A / ~ E
3	B	1 &E
4	~ B	1 &E
5	A & ~ A	2–4 ~ E

Derive: B & ~ B

1	A & ~ A	Assumption
2	~ (B & ~ B)	A / ~ E
3	A	1 &E
4	~ A	1 &E
5	B & ~ B	2–4 ~ E

c. Derive: (A ∨ B) ⊃ A

1	B ⊃ A	Assumption
2	A ∨ B	A / ⊃I
3	A	A / ∨E
4	A	3 R
5	B	A / ∨E
6	A	1, 5 ⊃E
7	A	2, 3–4, 5–6 ∨E
8	(A ∨ B) ⊃ A	2–7 ⊃I

Derive: B ⊃ A

```
 1 | (A ∨ B) ⊃ A                              Assumption
   |----------------
 2 | | B                                      A / ⊃I
   | |--------------
 3 | | A ∨ B                                  2 ∨I
 4 | | A                                      1, 3 ⊃E
 5 | B ⊃ A                                    2–4 ⊃I
```

e. Derive: ~ (A ≡ B)

```
 1 | (A & ~ B) ∨ (B & ~ A)                    Assumption
   |----------------------
 2 | | A & ~ B                                A / ∨E
   | |--------------------
 3 | | | A ≡ B                                A / ~ I
   | | |------------------
 4 | | | A                                    2 &E
 5 | | | B                                    3, 4 ≡E
 6 | | | ~ B                                  2 &E
 7 | | ~ (A ≡ B)                              2–6 ~ I
 8 | | B & ~ A                                A / ∨E
   | |--------------------
 9 | | | A ≡ B                                A / ~ I
   | | |------------------
10 | | | B                                    8 &E
11 | | | A                                    9, 10 ≡E
12 | | | ~ A                                  8 & E
13 | | ~ (A ≡ B)                              9–12 ~ I
14 | ~ (A ≡ B)                                1, 2–6, 7–13 ∨E
```

Derive: (A & ~ B) ∨ (B & ~ A)

```
 1 | ~ (A ≡ B)                                Assumption
   |------------------------------------
 2 | | ~ [(A & ~ B) ∨ (B & ~ A)]              A / ~ I
   | |----------------------------------
 3 | | | A                                    A / ≡I
   | | |--------------------------------
 4 | | | | ~ B                                A / ~ E
   | | | |------------------------------
 5 | | | | A & ~ B                            3, 4 &I
 6 | | | | (A & ~ B) ∨ (B & ~ A)              5 ∨I
 7 | | | | ~ [(A & ~ B) ∨ (B & ~ A)]          2 R
 8 | | | B                                    4–7 ~ I
 9 | | | B                                    A / ⊃I
   | | |--------------------------------
10 | | | | ~ A                                A / ~ E
   | | | |------------------------------
11 | | | | B & ~ A                            9, 10 &I
12 | | | | (A & ~ B) ∨ (B & ~ A)              11 ∨ I
13 | | | | ~ [(A & ~ B) ∨ (B & ~ A)]          2 R
14 | | | A                                    10–13 ~ E
15 | | A ≡ B                                  3–8, 9–14 ≡I
14 | | ~ (A ≡ B)                              1 R
15 | (A & ~ B) ∨ (B & ~ A)                    2–14 ~ E
```

5. Inconsistency

a. Derive: A ⊃ A, ~ (A ⊃ A)

1	~ (A ⊃ A)	Assumption
2	A	A / ⊃ I
3	A	2 R
4	A ⊃ A	2–3 ⊃I
5	~ (A ⊃ A)	1 R

c. Derive: A, ~ A

1	A ≡ B	Assumption
2	B ⊃ ~ A	Assumption
3	A	Assumption
4	A	3 R
5	B	1, 4 ≡E
6	~ A	2, 5 ⊃E

e. Derive: A, ~ A

1	A ⊃ ~ A	Assumption
2	~ A ⊃ A	Assumption
3	A	A / ~ I
4	~ A	1, 3 ⊃E
5	A	3 R
6	~ A	A / ~ I
7	A	2, 6 ⊃E

g. Derive: A ∨ B, ~ (A ∨ B)

1	~ (A ∨ B)	Assumption
2	C ⊃ A	Assumption
3	~ C ⊃ A	Assumption
4	C	A / ~ I
5	A	2, 4 ⊃E
6	A ∨ B	5 ∨I
7	~ (A ∨ B)	1 R
8	~ C	4–7 ~ I
9	B	3,8 ⊃E
10	A ∨ B	9 ∨I
11	~ (A ∨ B)	1 R

i. Derive: $F \vee G$, $\sim (F \vee G)$

1	$\sim (F \vee G) \equiv (A \supset A)$	Assumption
2	$H \supset F$	Assumption
3	$\sim H \supset F$	Assumption
4	│ A	A / $\supset$I
5	│ A	4 R
6	$A \supset A$	4–5 $\supset$ I
7	$\sim (F \vee G)$	1, 6 $\equiv$E
8	│ H	A / ~ I
9	│ F	2, 8 $\supset$E
10	│ $F \vee G$	9 $\vee$I
11	│ $\sim (F \vee G)$	7 R
12	$\sim H$	8–11 ~ I
13	F	3, 12 $\supset$E
14	$F \vee G$	13 $\vee$I

6. Derivability

a. Derive: $A \equiv B$

1	$A \supset B$	Assumption
2	$\sim A \supset \sim B$	Assumption
3	│ A	A / $\equiv$I
4	│ B	1, 3 $\supset$E
5	│ B	A / $\equiv$ I
6	│ │ $\sim A$	A / ~ E
7	│ │ $\sim B$	2, 6 $\supset$E
8	│ │ B	5 R
9	│ A	6–8 ~ E
10	$A \equiv B$	3–4, 5–9 $\equiv$I

c. Derive: A

1	$A \equiv (\sim B \vee C)$	Assumption
2	$B \supset C$	Assumption
3	│ $\sim A$	A / ~ E
4	│ │ B	A / ~ I
5	│ │ C	2, 4 $\supset$E
6	│ │ $\sim B \vee C$	5 $\vee$I
7	│ │ A	1, 6 $\equiv$E
8	│ │ $\sim A$	3 R
9	│ $\sim B$	4–8 ~ I
10	│ $\sim B \vee C$	9 $\vee$ I
11	│ A	1, 10 $\equiv$E
12	│ $\sim A$	3 R
13	A	3–12 ~ E

e. Derive: B ∨ D

1	│ B ∨ (C ∨ D)	Assumption
2	│ C ⊃ A	Assumption
3	│ A ⊃ ~ C	Assumption
4	│ │ B	A / ∨E
5	│ │ B ∨ D	4 ∨I
6	│ │ C ∨ D	A / ∨E
7	│ │ │ C	A / ∨E
8	│ │ │ │ ~ (B ∨ D)	A / ~ E
9	│ │ │ │ A	2, 7 ⊃E
10	│ │ │ │ ~ C	3, 9 ⊃E
11	│ │ │ │ C	7 R
12	│ │ │ B ∨ D	8–11 ~ E
13	│ │ │ D	A / ∨E
14	│ │ │ B ∨ D	13 ∨I
15	│ │ B ∨ D	6, 7–12, 13–14 ∨E
16	│ B ∨ D	1, 4–5, 6–15 ∨E

g. Derive: (A ∨ B) ⊃ ~ C

1	│ A ⊃ (D & B)	Assumption
2	│ (~ D ≡ B) & (C ⊃ A)	Assumption
3	│ │ A ∨ B	A / ⊃I
4	│ │ │ A	A / ∨E
5	│ │ │ │ C	A / ~ I
6	│ │ │ │ D & B	1, 4 ⊃E
7	│ │ │ │ ~ D ≡ B	2 &E
8	│ │ │ │ B	6 &E
9	│ │ │ │ ~ D	7, 8 ≡E
10	│ │ │ │ D	6 &E
11	│ │ │ ~ C	5–10 ~ I
12	│ │ │ B	A / ∨E
13	│ │ │ │ C	A / ~ I
14	│ │ │ │ C ⊃ A	2 &E
15	│ │ │ │ A	13, 14 ⊃E
16	│ │ │ │ D & B	1, 15 ⊃E
17	│ │ │ │ D	16 &E
18	│ │ │ │ ~ D ≡ B	2 &E
19	│ │ │ │ B	16 &E
20	│ │ │ │ ~ D	18, 19 ≡E
21	│ │ │ ~ C	13–20 ~ I
22	│ │ ~ C	3, 4–11, 12–21 ∨E
23	│ (A ∨ B) ⊃ ~ C	2–22 ⊃I

7. Validity

a. Derive: ~ (C ≡ ~ A)

1	│ ~ (C ∨ A)	Assumption
2	│ │ C ≡ ~ A	A / ~ I
3	│ │ │ ~ A	A / ~ E
4	│ │ │ C	2, 3 ≡E
5	│ │ │ C ∨ A	4 ∨I
6	│ │ │ ~ (C ∨ A)	1 R
7	│ │ A	3–6 ~ E
8	│ │ C ∨ A	7 ∨I
9	│ │ ~ (C ∨ A)	1 R
10	│ ~ (C ≡ ~ A)	2–9 ~ I

c. Derive: A ≡ B

1	│ ~ A & ~ B	Assumption
2	│ │ A	A / ≡I
3	│ │ │ ~ B	A / ~ E
4	│ │ │ ~ A	1 &E
5	│ │ │ A	2 R
6	│ │ B	3–5 ~ E
7	│ │ B	A / ≡I
8	│ │ │ ~ A	A / ~ E
9	│ │ │ ~ B	1 &E
10	│ │ │ B	7 R
11	│ │ A	8–10 ~ E
12	│ A ≡ B	2–6, 7–11 ≡I

e. Derive: ~ H

```
 1 | H ≡ ~ (I & ~ J)            Assumption
 2 | ~ I ≡ ~ H                  Assumption
 3 | J ⊃ ~ I                    Assumption
   |---------------
 4 |  | H                       A / ~ I
   |  |-----------
 5 |  | ~ (I & ~ J)             1, 4 ≡E
 6 |  |  | ~ I                  A / ~ E
   |  |  |--------
 7 |  |  | ~ H                  2, 6 ≡E
 8 |  |  | H                    4 R
 9 |  | I                       6–8 ~ E
10 |  |  | J                    A / ~ I
   |  |  |--------
11 |  |  | ~ I                  3, 10 ⊃E
12 |  |  | I                    9 R
13 |  | ~ J                     10–13 ~ I
14 |  | I & ~ J                 9, 13 & I
15 | ~ H                        4–14 ~ I
```

g. Derive: H ∨ ~ I

```
 1 | (F ∨ G) ∨ (H ∨ ~ I)        Assumption
 2 | F ⊃ H                      Assumption
 3 | I ⊃ ~ G                    Assumption
   |-------------------
 4 |  | F ∨ G                   A / ∨E
   |  |---------------
 5 |  |  | F                    A / ∨ E
   |  |  |-----------
 6 |  |  | H                    2, 5 ⊃E
 7 |  |  | H ∨ ~ I              6 ∨I
 8 |  |  | G                    A / ∨E
   |  |  |-----------
 9 |  |  |  | I                 A / ~ I
   |  |  |  |--------
10 |  |  |  | ~ G               3, 9 ⊃ E
11 |  |  |  | G                 8 R
12 |  |  | ~ I                  9–11 ~ I
13 |  |  | H ∨ ~ I              12 ∨I
14 |  | H ∨ ~ I                 4, 5–7, 8–13 ∨E
15 |  | H ∨ ~ I                 A / ∨E
   |  |---------------
16 |  | H ∨ ~ I                 15 R
17 | H ∨ ~ I                    1, 4–14, 15–16 ∨E
```

i. Derive: F ∨ (I & ~ G)

```
 1 │ ~ (F ∨ ~ G) ≡ ~ (H ∨ I)        Assumption
 2 │ F ∨ I                          Assumption
   ├──────────────────────
 3 │ │ F                            A / ∨E
   │ ├────────────────────
 5 │ │ F ∨ (I & ~ G)                3 ∨ I
 6 │ │ I                            A / ∨E
   │ ├────────────────────
 7 │ │ │ ~ (F ∨ ~ G)                A /~ E
   │ │ ├──────────────────
 8 │ │ │ ~ (H ∨ I)                  1, 7 ≡E
 9 │ │ │ H ∨ I                      6 ∨I
10 │ │ F ∨ ~ G                      7–9 ~ E
11 │ │ │ F                          A / ∨E
   │ │ ├──────────────────
12 │ │ │ F ∨ (I & ~ G)              11 ∨I
   │ │
13 │ │ │ ~ G                        A / ∨E
   │ │ ├──────────────────
14 │ │ │ I & ~ G                    6, 13 &I
15 │ │ │ F ∨ (I & ~ G)              15 ∨I
16 │ │ F ∨ (I & ~ G)                10, 11–12, 13–15 ∨E
17 │ F ∨ (I & ~ G)                  2, 3–5, 6–16 ∨E
```

k. Derive: (~ A ≡ ~ C) ⊃ (~ A ≡ D)

```
 1 │ (~ A ≡ ~ C) ≡ (B ≡ ~ D)        Assumption
 2 │ ~ A ⊃ ~ B                      Assumption
 3 │ C ⊃ ~ D                        Assumption
   ├──────────────────────
 4 │ │ ~ A ≡ ~ C                    A / ⊃I
   │ ├────────────────────
 5 │ │ │ ~ A                        A / ≡I
   │ │ ├──────────────────
 6 │ │ │ │ ~ D                      A / ~ E
   │ │ │ ├────────────────
 7 │ │ │ │ B ≡ ~ D                  1, 4 ⊃E
 8 │ │ │ │ B                        6, 7 ≡E
 9 │ │ │ │ ~ B                      2, 5 ⊃E
10 │ │ │ D                          6–9 ~ E
   │ │
11 │ │ │ D                          A / ≡I
   │ │ ├──────────────────
12 │ │ │ │ C                        A / ~ I
   │ │ │ ├────────────────
13 │ │ │ │ ~ D                      3, 12 ⊃E
14 │ │ │ │ D                        11 R
15 │ │ │ ~ C                        12–14 ~ I
16 │ │ │ ~ A                        4, 15 ≡E
17 │ │ ~ A ≡ D                      5–10, 11–16 ≡I
20 │ (~ A ≡ ~ C) ⊃ (~ A ≡ D)        4–17 ⊃I
```

m. Derive: $\sim E$

1	$\sim (A \supset B) \;\&\; (C \;\&\; \sim D)$	Assumption
2	$(B \lor \sim A) \lor [(C \;\&\; E) \supset D]$	Assumption
3	E	A / ~ I
4	$B \lor \sim A$	A / ∨E
5	B	A / ∨E
6	A	A / ⊃I
7	B	5 R
8	$A \supset B$	6–7 ⊃ I
9	$\sim A$	A / ∨E
10	A	A / ⊃I
11	$\sim B$	A / ~ E
12	A	10 R
13	$\sim A$	9 R
14	B	11–13 ~ E
15	$A \supset B$	10–14 ⊃I
16	$A \supset B$	4, 5–8, 9–15 ∨E
17	$(C \;\&\; E) \supset D$	A / ∨E
18	$\sim (A \supset B)$	A / ~ E
19	$C \;\&\; \sim D$	1 &E
20	$\sim D$	19 &E
21	C	19 &E
22	$C \;\&\; E$	3, 21 &I
22	D	17, 22 ⊃E
23	$A \supset B$	18–22 ~ E
24	$A \supset B$	2, 4–16, 17–23 ∨E
25	$\sim (A \supset B)$	1 &E
26	$\sim E$	3–25 ~ I

9. Theorems

a. Derive: $\sim (A \supset B) \supset \sim (A \equiv B)$

1	$\sim (A \supset B)$	A / ⊃I
2	$A \equiv B$	A / ~ I
3	A	A / ⊃I
4	B	2, 3 ≡E
5	$A \supset B$	3–4 ⊃I
6	$\sim (A \supset B)$	1 R
7	$\sim (A \equiv B)$	2–6 ~ I
8	$\sim (A \supset B) \supset \sim (A \equiv B)$	1–7 ⊃I

c. Derive: (A ⊃ B) ∨ (B ⊃ A)

```
 1  | | ~ [(A ⊃ B) ∨ (B ⊃ A)]          A / ~ E
    | |--------------------------
 2  | | | B                            A / ~ I
    | | |------------------------
 3  | | | | A                          A / ⊃I
    | | | |----------------------
 4  | | | | B                          2 R
 5  | | | A ⊃ B                        3–4 ⊃I
 6  | | | (A ⊃ B) ∨ (B ⊃ A)            5 ∨I
 7  | | | ~ [(A ⊃ B) ∨ (B ⊃ A)]        1 R
 8  | | ~ B                            2–7 ∨ I
 9  | | | B                            A / ⊃I
    | | |------------------------
10  | | | | ~ A                        A / ~ E
    | | | |----------------------
11  | | | | B                          9 R
12  | | | | ~ B                        8 R
13  | | | A                            10–12 ~ E
14  | | B ⊃ A                          9–13 ⊃I
15  | | (A ⊃ B) ∨ (B ⊃ A)              14 ∨I
16  | | ~ [(A ⊃ B) ∨ (B ⊃ A)]          1 R
17  | (A ⊃ B) ∨ (B ⊃ A)                1–16 ~ E
```

e. Derive: [(A ∨ B) ⊃ C] ≡ [(A ⊃ C) & (B ⊃ C)]

```
 1  | | (A ∨ B) ⊃ C                          A / ≡I
    | |--------------------------------
 2  | | | A                                  A / ⊃I
    | | |------------------------------
 3  | | | A ∨ B                              2 ∨I
 4  | | | C                                  1, 3 ⊃E
 5  | | A ⊃ C                                2–4 ⊃I
 6  | | | B                                  A / ⊃I
    | | |------------------------------
 7  | | | A ∨ B                              6 ∨I
 8  | | | C                                  1, 7 ⊃E
 9  | | B ⊃ C                                6–8 ⊃I
10  | | (A ⊃ C) & (B ⊃ C)                    5, 9 &I
11  | | (A ⊃ C) & (B ⊃ C)                    A / ≡I
    | |--------------------------------
12  | | | A ∨ B                              A / ⊃I
    | | |------------------------------
13  | | | | A                                A / ∨E
    | | | |----------------------------
14  | | | | A ⊃ C                            11 &E
15  | | | | C                                13, 14 ⊃E
16  | | | | B                                A / ∨E
    | | | |----------------------------
17  | | | | B ⊃ C                            11 &E
18  | | | | C                                16, 17 ⊃E
19  | | | C                                  12, 13–15, 16–18 ∨E
20  | | (A ∨ B) ⊃ C                          12–19 ⊃I
21  | [(A ∨ B) ⊃ C] ≡ [(A ⊃ C) & (B ⊃ C)]    1–10, 11–20 ≡I
```

g. Derive: ~ (A ≡ B) ≡ (A ≡ ~ B)

1	│ │ ~ (A ≡ B)	A / ≡I
2	│ │ │ A	A / ≡I
3	│ │ │ │ B	A / ~ I
4	│ │ │ │ │ A	A / ≡I
5	│ │ │ │ │ B	3 R
6	│ │ │ │ │ B	A / ≡I
7	│ │ │ │ │ A	2 R
8	│ │ │ │ A ≡ B	4–5, 6–7 ≡I
9	│ │ │ │ ~ (A ≡ B)	1 R
10	│ │ │ ~ B	3–9 ~ I
11	│ │ │ ~ B	A / ≡I
12	│ │ │ │ ~ A	A / ~ E
13	│ │ │ │ │ A	A / ≡I
14	│ │ │ │ │ │ ~ B	A / ~ E
15	│ │ │ │ │ │ A	13 R
16	│ │ │ │ │ │ ~ A	12 R
17	│ │ │ │ │ B	14–16 ~ E
18	│ │ │ │ │ B	A / ≡I
19	│ │ │ │ │ │ ~ A	A / ~ E
20	│ │ │ │ │ │ B	18 R
21	│ │ │ │ │ │ ~ B	11 R
22	│ │ │ │ │ A	19–21 ~ E
23	│ │ │ │ A ≡ B	13–17, 18–22
24	│ │ │ │ ~ (A ≡ B)	1 R
25	│ │ │ A	12–24 ~ E
26	│ │ A ≡ ~ B	2–10, 11–25 ≡I
27	│ │ A ≡ ~ B	A / ≡I
28	│ │ │ A ≡ B	A / ~ I
29	│ │ │ │ B	A / ~ I
30	│ │ │ │ A	28, 29 ≡E
31	│ │ │ │ ~ B	27, 30 ≡E
32	│ │ │ │ B	29 R
33	│ │ │ ~ B	29–32 ~ I
34	│ │ │ │ ~ B	A / ~ E
35	│ │ │ │ A	27, 34 ≡E
36	│ │ │ │ B	28, 35 ≡E
37	│ │ │ │ ~ B	34 R
38	│ │ │ B	34–37 ~ E
39	│ │ ~ (A ≡ B)	28–38 ~ I
40	│ ~ (A & B)	

10. Equivalence

a. Derive: ~ ~ A

1	A	Assumption
2	~ A	A / ~ I
3	A	1 R
4	~ A	2 R
5	~ ~ A	2–4 ~ I

Derive: A

1	~ ~ A	Assumption
2	~ A	A / ~ E
3	~ A	2 R
4	~ ~ A	1 R
5	A	2–4 ~ E

c. Derive: A ∨ A

1	A	Assumption
2	A ∨ A	A / ∨I

Derive: A ∨ A

1	A ∨ A	Assumption
2	A	A / ∨E
3	A	2 R
4	A	1, 2–3, 2–3 ∨E

e. Derive: B ∨ A

1	A ∨ B	Assumption
2	A	A / ∨E
3	B ∨ A	2 ∨I
4	B	A / ∨E
5	B ∨ A	4 ∨I
6	B ∨ A	1, 2–3, 4–5 ∨E

Derive: $A \vee B$

1	$B \vee A$	Assumption
2	B	A / $\vee$E
3	$A \vee B$	2 $\vee$I
4	A	A / $\vee$E
5	$A \vee B$	4 $\vee$I
6	$A \vee B$	1, 2–3, 4–5 $\vee$E

g. Derive: $(A \vee B) \vee C$

1	$A \vee (B \vee C)$	Assumption
2	A	A / $\vee$E
3	$A \vee B$	2 $\vee$I
4	$(A \vee B) \vee C$	3 $\vee$I
5	$B \vee C$	A / $\vee$E
6	B	A / $\vee$E
7	$A \vee B$	6 $\vee$I
9	$(A \vee B) \vee C$	7 $\vee$I
10	C	A / $\vee$I
11	$(A \vee B) \vee C$	10 $\vee$I
12	$(A \vee B) \vee C$	5, 6–9, 10–11 $\vee$I
13	$(A \vee B) \vee C$	1, 2–4, 5–12 $\vee$E

Derive: $A \vee (B \vee C)$

1	$(A \vee B) \vee C$	Assumption
2	$A \vee B$	A / $\vee$E
3	A	A / $\vee$E
4	$A \vee (B \vee C)$	3 $\vee$I
5	B	A / $\vee$E
6	$B \vee C$	5 $\vee$I
7	$A \vee (B \vee C)$	6 $\vee$I
8	$A \vee (B \vee C)$	2, 3–4, 5–7 $\vee$E
9	C	A / $\vee$E
10	$B \vee C$	9 $\vee$I
11	$A \vee (B \vee C)$	10 $\vee$I
12	$A \vee (B \vee C)$	1, 2–8, 9–11 $\vee$E

i. Derive: ~ B ⊃ ~ A

1	│ A ⊃ B	Assumption
2	│ │ ~ B	A / ⊃I
3	│ │ │ A	A / ~ E
4	│ │ │ B	1, 3 ⊃E
5	│ │ │ ~ B	2 R
6	│ │ ~ A	3–5 ~ I
7	│ ~ B ⊃ ~ A	2–6 ⊃I

Derive: A ⊃ B

1	│ ~ B ⊃ ~ A	Assumption
2	│ │ A	A / ⊃I
3	│ │ │ ~ B	A / ~ E
4	│ │ │ A	2 R
5	│ │ │ ~ A	1, 3 ⊃E
6	│ │ B	3–5 ~ E
7	│ A ⊃ B	2–6 ⊃I

k. Derive: A ≡ B

1	│ (A & B) ∨ (~ A & ~ B)	Assumption
2	│ │ A & B	A / ∨E
3	│ │ │ A	A / ≡I
4	│ │ │ B	2 &E
5	│ │ │ B	A / ≡I
6	│ │ │ A	2 &E
7	│ │ A ≡ B	3–4, 5–6 ≡I
8	│ │ ~ A & ~ B	A / ∨E
9	│ │ │ A	A / ≡I
10	│ │ │ │ ~ B	A / ~ E
11	│ │ │ │ A	9 R
12	│ │ │ │ ~ A	8 &E
13	│ │ │ B	10–12 ~ E
14	│ │ │ B	A / ≡I
15	│ │ │ │ ~ A	A / ~ E
16	│ │ │ │ B	14 R
17	│ │ │ │ ~ B	8 &E
18	│ │ │ A	15–17 ~ E
19	│ │ A ≡ B	9–13, 14–18 ≡I
20	│ A ≡ B	1, 2–7, 8–19 ∨E

Derive: (A & B) ∨ (~ A & ~ B)

1	A ≡ B	Assumption
2	~ [(A & B) ∨ (~ A & ~ B)]	A / ~ E
3	A	A / ~ I
4	B	1, 3 ≡E
5	A & B	3, 4 &I
6	(A & B) ∨ (~ A & ~ B)	5 ∨ I
7	~ [(A & B) ∨ (~ A & ~ B)]	2 R
8	~ A	3–7 ~ I
9	B	A / ~ I
10	A	1, 9 ≡E
11	~ A	8 R
12	~ B	9–11 ~ I
13	~ A & ~ B	8, 12 &I
14	(A & B) ∨ (~ A & ~ B)	13 ∨I
15	~ [(A & B) ∨ (~ A & ~ B)]	2 R
16	(A & B) ∨ (~ A & ~ B)	2–15 ~ E

m. Derive: (A ∨ B) & (A ∨ C)

1	A ∨ (B & C)	Assumption
2	A	A / ∨E
3	A ∨ B	2 ∨I
4	A ∨ C	2 ∨I
5	(A ∨ B) & (A ∨ C)	3, 4 &I
6	B & C	A / ∨E
7	B	6 & E
8	A ∨ B	7 ∨I
9	C	6 &E
10	A ∨ C	9 ∨I
11	(A ∨ B) & (A ∨ C)	8, 10 &I
12	(A ∨ B) & (A ∨ C)	1, 2–5, 6–11 ∨E

Derive: A ∨ (B & C)

```
 1  | (A ∨ B) & (A ∨ C)          Assumption
    |-------------------
 2  | A ∨ B                      1 &E
 3  | | A                        A / ∨E
    | |-----------------
 4  | | A ∨ (B & C)              3 ∨I
    |
 5  | | B                        A / ∨E
    | |-----------------
 6  | | A ∨ C                    1 &E
 7  | | | A                      A / ∨E
    | | |---------------
 8  | | | A ∨ (B & C)            7 ∨I
    | |
 9  | | | C                      A / ∨E
    | | |---------------
10  | | | B & C                  5, 9 &I
11  | | | A ∨ (B & C)            10 ∨I
12  | | A ∨ (B & C)              6, 7–8, 9–11 ∨E
13  | A ∨ (B & C)                2, 3–4, 5–12 ∨E
```

o. Derive: ~ A ∨ ~ B

```
 1  | ~ (A & B)                  Assumption
    |-------------------
 2  | | ~ (~ A ∨ ~ B)            A / ~ E
    | |-----------------
 3  | | | ~A                     A / ~ E
    | | |---------------
 4  | | | ~ A ∨ ~ B              3 ∨I
 5  | | | ~ (~ A ∨ ~ B)          2 R
 6  | | A                        3–5 ~ E
 7  | | | ~ B                    A / ~ E
    | | |---------------
 8  | | | ~ A ∨ ~ B              7 ∨I
 9  | | | ~ (~ A ∨ ~ B)          2 R
10  | | B                        7–9 ~ E
11  | | A & B                    6, 10 &I
12  | | ~ (A & B)                1 R
13  | ~ A ∨ ~ B                  2–12 ~ E
```

Derive: ~ (A & B)

```
 1 | ~ A ∨ ~ B                  Assumption
   |----------
 2 |   | A & B                  A / ~ I
   |   |------
 3 |   |   | ~ A              A / ∨E
   |   |   |----
 4 |   |   | ~ A              3 R

 5 |   |   | ~ B              A / ∨E
   |   |   |----
 6 |   |   |   | A            A / ~ I
   |   |   |   |--
 7 |   |   |   | B            2 &E
 8 |   |   |   | ~ B          5 R
 9 |   |   | ~ A              6–8 ~ I
10 |   | ~ A                    1, 3–4, 5–9 ∨E
11 |   | A                      2 &E
12 | ~ (A & B)                  2–11 ~ I
```

12. Inconsistency

a. Derive: B, ~ B

```
 1 | (A ⊃ B) & (A ⊃ ~ B)        Assumption
 2 | (C ⊃ A) & (~ C ⊃ A)        Assumption
   |--------------------
 3 | A ⊃ B                      1 &E
 4 | A ⊃ ~ B                    1 &E
 5 |   | C                      A / ~ I
   |   |----------------
 6 |   | C ⊃ A                  2 &E
 7 |   | A                      5, 6 ⊃E
 8 |   | B                      3, 7 ⊃E
 9 |   | ~ B                    4, 7 ⊃E
10 | ~ C                        5–9 ~ I
11 | ~ C ⊃ A                    2 &E
12 | A                          10, 11 ⊃E
13 | B                          3, 12 ⊃E
14 | ~ B                        4, 12 ⊃E
```

c. Derive: A, ~ A

1	│ C ≡ ∨ A	Assumption
2	│ C ≡ A	Assumption
3	│ │ A	A / ~ I
4	│ │ C	2, 3 ≡E
5	│ │ ~ A	1, 4 ≡E
6	│ │ A	3 R
7	│ ~A	3–6 ~ I
8	│ │ ~A	A / ~ E
9	│ │ C	1, 8 ≡E
10	│ │ A	2, 9 ≡E
11	│ │ ~ A	8 R
12	│ A	8–11 ~ E

e. Derive: A, ~ A

1	│ ~ [(A ∨ B) ∨ C]	Assumption
2	│ A ≡ ~ C	Assumption
3	│ │ A	A / ~ 1
4	│ │ A ∨ B	3 ∨I
5	│ │ (A ∨ B) ∨ C	4 ∨I
6	│ │ ~ [(A ∨ B) ∨ C]	1 R
7	│ ~A	3–6 ~ I
8	│ │ ~ A	A / ~ E
9	│ │ │ C	A / ~ I
10	│ │ │ A	2, 9 ≡E
11	│ │ │ ~ A	8 R
12	│ │ ~ C	9–11 ~ I
13	│ │ A	2, 12 ≡E
14	│ │ ~ A	8 R
15	│ A	8–14 ~ E

g. Derive: B, ~ B

1	A & (B ∨ C)	Assumption
2	(~ C ∨ H) & (H ⊃ ~ H)	Assumption
3	~ B	Assumption
4	B ∨ C	1 &E
5	B	A / ∨E
6	B	5 R
7	C	A / ∨E
8	~ C ∨ H	2 &E
9	~ C	A / ∨E
10	~ B	A / ~ E
11	C	7 R
12	~ C	9 R
13	B	10–12 ~ E
14	H	A / ∨E
15	~ B	A / ~ E
16	H ⊃ ~ H	2 &E
17	~ H	14, 16 ⊃E
18	B	15, 17 ~ E
19	B	8, 9–13, 14–18 ∨E
20	B	4, 5–6, 7–19 ∨E
21	~ B	3 R

13. Validity

a. Derive: M

1	S & F	Assumption
2	F ⊃ B	Assumption
3	(B & ~ M) ⊃ ~ S	Assumption
4	~ M	A / ~ E
5	F	1 &E
6	B	2, 5 ⊃E
7	B & ~ M	6, 4 &I
8	~ S	3, 7 ⊃E
9	S	1 &E
10	M	4–9 ~ E

c. Derive: ~ J

1	(C ⊃ ~ R) & (R ⊃ L)	Assumption
2	C ≡ (C ∨ L)	Assumption
3	J ⊃ R	Assumption
4	J	A / ~ I
5	R	3, 4 ⊃E
6	R ⊃ L	1 &E
7	L	5, 6 ⊃E
8	C ∨ L	7 ∨I
9	C	2, 8 ≡E
10	C ⊃ ~ R	1 &E
11	~ R	9, 10 ⊃E
12	~ J	4–11 ~ I

e. Derive: ~ M

1	~ (R ∨ W)	Assumption
2	(R ≡ M) ∨ [(M ∨ G) ⊃ (W ≡ M)]	Assumption
3	M	A / ~ I
4	R ≡ M	A / ∨E
5	R	3, 4 ≡E
6	R ∨ W	5 ∨I
7	(M ∨ G) ⊃ (W ≡ M)	A / ∨E
8	M ∨ G	3 ∨I
9	W ≡ M	7, 8 ⊃E
10	W	3, 9 ≡E
11	R ∨ W	10 ∨I
12	R ∨ W	2, 4–6, 7–11 ∨E
13	~ (R ∨ W)	1 R
14	~ M	3–13 ~ I

g. Derive: H ⊃ J

1	(H & T) ⊃ J	Assumption
2	(M ⊃ D) & (~ D ⊃ M)	Assumption
3	~ T ≡ (~ D & M)	Assumption
4	H	A / ⊃I
5	~ J	A / ~ E
6	T	A / ~ I
7	H & T	4, 6 &I
8	J	1, 7 ⊃E
9	~ J	5 R
10	~ T	6–9 ~ I
11	~ D & M	3, 10 ≡E
12	M ⊃ D	2 &E
13	M	11 &E
14	D	12, 13 ⊃E
15	~ D	11 &E
16	J	5–15 ~ E
17	H ⊃ J	4–16 ⊃I

i. Derive: L ⊃ T

```
 1  | L ⊃ (C ∨ T)                          Assumption
 2  | (~ L ∨ B) & (~ B ∨ ~ C)              Assumption
    |------------------------
 3  | | L                                  A / ⊃I
    | |----------------------
 4  | | C ∨ T                              1, 3 ⊃E
 5  | | | C                                A / ∨E
    | | |--------------------
 6  | | | ~ B ∨ ~ C                        2 &E
 7  | | | | ~ B                            A / ∨E
    | | | |------------------
 8  | | | | ~ L ∨ B                        2 &E
 9  | | | | | ~ L                          A / ∨E
    | | | | |----------------
10  | | | | | | ~ T                        A / ~ E
    | | | | | |--------------
11  | | | | | | L                          3 R
12  | | | | | | ~ L                        9 R
13  | | | | | T                            10–12 ~ E
14  | | | | | B                            A / ∨E
    | | | | |----------------
15  | | | | | | ~ T                        A / ~ E
    | | | | | |--------------
16  | | | | | | B                          14 R
17  | | | | | | ~ B                        7 R
18  | | | | | T                            15–17 ~ E
19  | | | | T                              8, 9–13, 14–18 ∨E
20  | | | | ~ C                            A / ∨E
    | | | |------------------
21  | | | | | ~ T                          A / ~ E
    | | | | |----------------
22  | | | | | ~ C                          20 R
23  | | | | | C                            5 R
24  | | | | T                              21–23 ~ E
25  | | | T                                6, 7–19, 20–24 ∨E
26  | | | T                                A / ∨E
    | | |--------------------
27  | | | T                                26 R
28  | | T                                  4, 5–25, 26–27 ∨E
29  | L ⊃ T                                3–28 ⊃ I
```

14. Inconsistency

a.

1	(M ⊃ B) & (B ⊃ P)	Assumption
2	M & ~ P	Assumption
3	M	2 &E
4	M ⊃ B	1 &E
5	B	3, 4 ⊃E
6	B ⊃ P	1 &E
7	P	5, 6 ⊃E
8	~ P	2 &E

c.

1	B ⊃ I	Assumption
2	(~ B & ~ I) ⊃ C	Assumption
3	~ C & ~ I	Assumption
4	│ B	A / ~ I
5	│ I	1, 4 ⊃E
6	│ ~ I	3 &E
7	~ B	4–6 ~ I
8	~ I	3 &E
9	~ B & ~ I	7, 8 &I
10	C	2, 9 ⊃E
11	~ C	3 &E

e.

1	M ∨ (F ⊃ T)	Assumption
2	N ≡ ~ T	Assumption
3	(F & N) & ~ M	Assumption
4	│ M	A / ∨E
5	│ M	4 R
6	│ F ⊃ T	A / ∨E
7	│ │ ~ M	A / ~ E
8	│ │ F & N	3 &E
9	│ │ F	8 &E
10	│ │ T	6, 9 ⊃E
11	│ │ N	8 &E
12	│ │ ~ T	2, 11 ≡E
13	│ M	7–12 ~ E
14	M	1, 4–5, 6–13 ∨E
15	~ M	3 &E

15.a. We do not want this rule as a rule of *SD* because it is not truth-preserving. The truth of **P** ∨ **Q** does not entail the truth of **P**.

c. We can show that Reiteration is dispensable by explaining how do derive **P** whenever **P** occurs on an earlier accessible line, without using Reiteration. Assume that **P** occurs on an accessibleline **i** and that we want to derive **P** on a later line. We can do this as follows:

i	**P**	…
n	**P** & **P**	**i**, **i** &I
n + 1	**P**	**n** &E

e. Assume that **P** is a theorem is *SD*. Now considere any argument has ~ P as one of its premises:

$\mathbf{Q}_1$
$\mathbf{Q}_2$
….
$\mathbf{Q}_n$
~ **P**

R

We can derive **R** from the set consisting of the premises as follows:

1	**Q1**		Assumption
2	**Q2**		Assumption
	…		
n	**Qn**		Assumption
n + 1	~ **P**		Assumption
n + 2		~ **R**	A / ~ E
n + 3		~ **P**	
n + 4			**n** + 1 R
		…	
n + **k**		**P**	
n + **k** + 1	**R**		**n** + 2 − **n** + **k** ~ E

Here lines **n** + 4 through **n** + **k** consists of the derivation of **P** from no primary assumptions. We know there is such a derivation because we know **P** is a theorem of *SD*.

e. If **P** is a theorem is *SD* then any argument of *SL* that has ~ **P** among its premises is vald in *SD*. We can construct a derivation of the conclusion, call it **Q** by taking the premises of the argument as our primary assumptions.

~ **P** will be one of these assumptions. Next assume ~ **Q**, derive both **P** and ~ **P**, and obtain **Q** by Negation Elimination. We can obtain ~ **P** by Reiteration since it is one of the primary assumptions of the derivation. We can obtain **P** because it is a theorem of *SD* and therefore can be derived from the empty set. If it can be derived from the empty set it can also be derived from the set consisting of the premises of the argument, by inserting the derivation of **P** from the empty set within the scope of the assumption ~ **Q**.

16. We here make use of a result established in Sections 6.3 and 6.4,

$\Gamma \vdash \mathbf{P}$ in *SD* if and only if $\Gamma \vDash \mathbf{P}$

a. Assume that a given argument is valid in *SD*. Then we know that its conclusion is derivable in *SD* from the set consisting of its premises. By the above result it follows that the conclusion of the argument is truth-functionally entailed by the set consisting of the premises of the argument. Therefore there is no truth-value assignment on which the members of the set, which are just the premises of the argument, are true and the conclusion of the argument false. So the argument is truth-functionally valid. Conversely, assume that the given argument is truth-functionally valid. So there is no truth-value assignment on which the premises of the argument are true and the conclusion false. From this it follows that the set consisting of the premises of the argument truth-functionally entails the conclusion of the argument. And by the above result it next follows that the conclusion of the argument is derivable from the set consisting of the premises of the argument, and from this it follows that the argument is valid in *SD*.

d. Assume that sentences **P** and **Q** of *SL* are equivalent in *SD*. Then each can be derived from the unit set of the other. By the above result it follows that the unit set of each truth-functionally entails the other. So there is no truth-value assignment on which **P** is true and **Q** false, and no truth-value assignment on which **Q** is true and **P** false. So **P** and **Q** are truth-functionally equivalent.

Exercises 5.4E

Derive: ~ D

1	D ⊃ E	Assumption
2	E ⊃ (Z & W)	Assumption
3	~ Z ∨ ~ W	Assumption
4	~ (Z & W)	3 DeM
5	~ E	2, 4 MT
6	~ D	1, 5 MT

c. Derive: K

1	(W ⊃ S) & ~ M	Assumption
2	(~ W ⊃ H) ∨ M	Assumption
3	(~ S ⊃ H) ⊃ K	Assumption
4	W ⊃ S	1 &E
5	~ S ⊃ ~ W	4 Trans
6	~ M	1 &E
7	~ W ⊃ H	2, 6 DS
8	~ S ⊃ H	5, 7 HS
9	K	3, 8 ⊃E

e. Derive: C

1	(M ∨ B) ∨ (C ∨ G)	Assumption
2	~ B & (~ G & ~ M)	Assumption
3	~ B	2 &E
4	(B ∨ M) ∨ (C ∨ G)	1 Com
5	B ∨ [M ∨ (C ∨ G)]	4 Assoc
6	M ∨ (C ∨ G)	3, 5 DS
7	~ G & ~ M	2 &E
8	~ G	7 &E
9	(M ∨ C) ∨ G	6 Assoc
10	M ∨ C	8, 9 DS
11	~ M	7 &E
12	C	10, 11 DS

2. Validity

a. Derive: Y ≡ Z

1	~ Y ⊃ ~ Z	Assumption
2	~ Z ⊃ ~ X	Assumption
3	~ X ⊃ ~ Y	Assumption
4	Y	A / ≡E
5	~ Z ⊃ ~ Y	2, 3 HS
6	Y ⊃ Z	5 Trans
7	Z	4, 6 ⊃E
8	Z	A / ≡I
9	Z ⊃ Y	1 Trans
10	Y	8, 9 ⊃E
11	Y ≡ Z	4–7, 8–10 ≡I

c. Derive: I ⊃ ~ D

```
 1 | (F & G) ∨ (H & ~ I)            Assumption
 2 | I ⊃ ~ (F & D)                  Assumption
   |----------------------
 3 |   | I                          A / ⊃I
   |   |------------------
 4 |   | ~ (F & D)                  2, 3 ⊃E
 5 |   | ~ F ∨ ~ D                  4 DeM
 6 |   | ~ ~ I                      3 DN
 7 |   | ~ H ∨ ~ ~ I                6 ∨I
 8 |   | ~ (H & ~ I)                7 DeM
 9 |   | F & G                      1, 8 DS
10 |   | F                          9 &E
11 |   | ~ ~ F                      10 DN
12 |   | ~ D                        5, 11 DS
13 | I ⊃ ~ D                        3–12 ⊃I
```

e. Derive: I ∨ H

```
 1 | F ⊃ (G ⊃ H)                    Assumption
 2 | ~ I ⊃ (F ∨ H)                  Assumption
   | F ⊃ G                          Assumption
 3 |----------------
 4 |   | ~ I                        A / ⊃I
   |   |------------
 5 |   | F ∨ H                      2, 4 ⊃E
 6 |   |   | ~ H                    A / ~ E
   |   |   |--------
 7 |   |   | F                      5, 6 DS
 8 |   |   | G                      3, 7 ⊃E
 9 |   |   | G ⊃ H                  1, 7 ⊃E
10 |   |   | ~ G                    6, 9 MT
11 |   | H                          6–10 ~ E
12 | ~ I ⊃ H                        4–11 ⊃I
13 | ~ ~ I ∨ H                      12 Impl
14 | I ∨ H                          13 DN
```

g. Derive: $X \equiv Y$

1	$\mid [(X \mathbin{\&} Z) \mathbin{\&} Y] \vee (\sim X \supset \sim Y)$	Assumption
2	$\mid X \supset Z$	Assumption
3	$\mid Z \supset Y$	Assumption
4	$\mid \; \mid X$	A / $\equiv$I
5	$\mid \; \mid Z$	2, 4 $\supset$E
6	$\mid \; \mid Y$	3, 5 $\supset$E
7	$\mid \; \mid Y$	A / $\equiv$I
8	$\mid \; \mid \; \mid (X \mathbin{\&} Z) \mathbin{\&} Y$	A / $\vee$E
9	$\mid \; \mid \; \mid X \mathbin{\&} Z$	8 &E
10	$\mid \; \mid \; \mid X$	9 &E
11	$\mid \; \mid \; \mid \sim X \supset \sim Y$	A / $\vee$E
12	$\mid \; \mid \; \mid Y \supset X$	11 Trans
13	$\mid \; \mid \; \mid X$	7, 12 $\supset$E
14	$\mid \; \mid X$	1, 8–10, 11–13 $\vee$E
15	$\mid X \equiv Y$	4–6, 7–14 $\equiv$I

3. Theorems

a. Derive: $A \vee \sim A$

1	$\mid \; \mid \sim (A \vee \sim A)$	A / $\sim$ E
2	$\mid \; \mid \sim A \mathbin{\&} \sim \sim A$	1 DeM
3	$\mid \; \mid \sim A$	2 &E
4	$\mid \; \mid \sim \sim A$	2 &E
5	$\mid A \vee \sim A$	1–4 $\sim$ E

c. Derive: $A \vee [(\sim A \vee B) \mathbin{\&} (\sim A \vee C)]$

1	$\mid \; \mid \sim A$	A / $\supset$I
2	$\mid \; \mid \sim A \vee (B \mathbin{\&} C)$	1 $\vee$I
3	$\mid \; \mid (\sim A \vee B) \mathbin{\&} (\sim A \vee C)$	2 Dist
4	$\mid \sim A \supset [(\sim A \vee B) \mathbin{\&} (\sim A \vee C)]$	1–3 $\supset$I
5	$\mid \sim \sim A \vee [(\sim A \vee B) \mathbin{\&} (\sim A \vee C)]$	4 Impl
6	$\mid A \vee [(\sim A \vee B) \mathbin{\&} (\sim A \vee C)]$	5 DN

e. Derive: [A ⊃ (B & C)] ≡ [(~ B ∨ ~ C) ⊃ ~ A]

1	A ⊃ (B & C)	A / ≡I
2	~ (B & C) ⊃ ~ A	1 Trans
3	(~ B ∨ ~ C) ⊃ ~ A	2 DeM
4	(~ B ∨ ~ C) ⊃ ~ A	A / ≡I
5	~ (B & C) ⊃ ~ A	4 DeM
6	A ⊃ (B & C)	5 Trans
7	[A ⊃ (B & C)] ≡ [(~ B ∨ ~ C) ⊃ ~ A]	1–3, 4–6 ≡I

g. Derive: [A ⊃ (B ≡ C)] ≡ (A ⊃ [(~ B ∨ C) & (~ C ∨ B)])

1	A ⊃ (B ≡ C)	A / ≡I
2	A ⊃ [(B ⊃ C) & (C ⊃ B)]	1 Equiv
3	A ⊃ [(~ B ∨ C) & (C ⊃ B)]	2 Impl
4	A ⊃ [(~ B ∨ C) & (~ C ∨ B)]	3 Impl
5	A ⊃ [(~ B ∨ C) & (~ C ∨ B)]	A / ≡I
6	A ⊃ [(B ⊃ C) & (~ C ∨ B)]	5 Impl
7	A ⊃ [(B ⊃ C) & (C ⊃ B)]	6 Impl
8	A ⊃ (B ≡ C)	7 Equiv
9	[A ⊃ (B ≡ C)] ≡ (A ⊃ [(~ B ∨ C) & (~ C ∨ B)])	1–4, 5–8 ≡I

i. Derive: [~ A ⊃ (~ B ⊃ C)] ⊃ [(A ∨ B) ∨ (~ ~ B ∨ C)]

1	~ A ⊃ (~ B ⊃ C)	A / ⊃I
2	~ ~ A ∨ (~ B ⊃ C)	1 Impl
3	~ ~ A ∨ (~ ~ B ∨ C)	2 Impl
4	A ∨ (~ ~ B ∨ C)	3 DN
5	A ∨ [(~ ~ B ∨ ~ ~ B) ∨ C]	4 Idem
6	A ∨ [~ ~ B ∨ (~ ~ B ∨ C)]	5 Assoc
7	(A ∨ ~ ~ B) ∨ (~ ~ B ∨ C)	6 Assoc
8	(A ∨ B) ∨ (~ ~ B ∨ C)	7 DN
9	[~ A ⊃ (~ B ⊃ C)] ⊃ [(A ∨ B) ∨ (~ ~ B ∨ C)]	1–8 ⊃I

4. Equivalence

a. Derive: ~ (~ A & ~ B)

1	A ∨ B	Assumption
2	~ ~ A ∨ B	1 DN
3	~ ~ A ∨ ~ ~ B	2 DN
4	~ (~ A & ~ B)	3 DeM

Derive: $A \vee B$

1	$\sim (\sim A \,\&\, \sim B)$	Assumption
2	$\sim \sim A \vee \sim \sim B$	1 DeM
3	$A \vee \sim \sim B$	2 DN
4	$A \vee B$	3 DN

c. Derive: $\sim (A \supset C) \supset \sim B$

1	$(A \,\&\, B) \supset C$	Assumption
2	$(B \,\&\, A) \supset C$	1 Com
3	$B \supset (A \supset C)$	2 Exp
4	$\sim (A \supset C) \supset \sim B$	3 Trans

Derive: $(A \,\&\, B) \supset C$

1	$\sim (A \supset C) \supset \sim B$	Assumption
2	$B \supset (A \supset C)$	1 Trans
3	$(B \,\&\, A) \supset C$	2 Exp
4	$(A \,\&\, B) \supset C$	3 Com

e. Derive: $A \vee (\sim B \equiv \sim C)$

1	$A \vee (B \equiv C)$	Assumption
2	$A \vee [(B \supset C) \,\&\, (C \supset B)]$	1 Equiv
3	$A \vee [(\sim C \supset \sim B) \,\&\, (C \supset B)]$	2 Trans
4	$A \vee [(\sim C \supset \sim B) \,\&\, (\sim B \supset \sim C)]$	3 Trans
5	$A \vee [(\sim B \supset \sim C) \,\&\, (\sim C \supset \sim B)]$	4 Com
6	$A \vee (\sim B \equiv \sim C)$	5 Equiv

Derive: $A \vee (B \equiv C)$

1	$A \vee (\sim B \equiv \sim C)$	Assumption
2	$A \vee [(\sim B \supset \sim C) \,\&\, (\sim C \supset \sim B)]$	1 Equiv
3	$A \vee [(C \supset B) \,\&\, (\sim C \supset \sim B)]$	2 Trans
4	$A \vee [(C \supset B) \,\&\, (B \supset C)]$	3 Trans
5	$A \vee [(B \supset C) \,\&\, (C \supset B)]$	4 Com
6	$A \vee (B \equiv C)$	5 Equiv

5. Inconsistency

a. 1	$[(E \,\&\, F) \vee \sim \sim G] \supset M$	Assumption
2	$\sim [[(G \vee E) \,\&\, (F \vee G)] \supset (M \,\&\, M)]$	Assumption
3	$\sim ([(G \vee E) \,\&\, (F \vee G)] \supset M)$	2 Idem
4	$\sim ([(G \vee E) \,\&\, (G \vee F)] \supset M)$	3 Com
5	$\sim ([G \vee (E \,\&\, F)] \supset M)$	4 Dist
6	$\sim ([(E \,\&\, F) \vee G] \supset M)$	5 Com
7	$\sim ([(E \,\&\, F) \vee \sim \sim G] \supset M)$	6 DN

c.	1	M & L	Assumption
	2	[L & (M & ~ S)] ⊃ K	Assumption
	3	~ K ∨ ~ S	Assumption
	4	~ (K ≡ ~ S)	Assumption
	5	K ⊃ ~ S	3 Impl
	6	[(L & M) & ~ S] ⊃ K	2 Assoc
	7	(L & M) ⊃ (~ S ⊃ K)	6 Exp
	8	L & M	1 Com
	9	~ S ⊃ K	7, 8 ⊃E
	10	(K ⊃ ~ S) & (~ S ⊃ K)	5, 9 &I
	11	K ≡ ~ S	10 Equiv

e.	1	~ [W & (Z ∨ Y)]	Assumption
	2	(Z ⊃ Y) ⊃ Z	Assumption
	3	(Y ⊃ Z) ⊃ W	Assumption
	4	~ W ∨ ~ (Z ∨ Y)	1 DeM
	5	│ ~ Z	A / ~ E
	6	│ ~ (Z ⊃ Y)	2, 5 MT
	7	│ ~ (~ Z ∨ Y)	6 Impl
	8	│ ~ ~ Z & ~ Y	7 DeM
	9	│ ~ ~ Z	8 &E
	10	│ ~ Z	5 R
	11	Z	5–10 ~ E
	12	Z ∨ Y	11 ∨I
	13	~ ~ (Z ∨ Z)	12 DN
	14	~ W	4, 13 DS
	15	~ (Y ⊃ Z)	3, 14 MT
	16	~ (~ Y ∨ Z)	15 Impl
	17	~ ~ Y & ~ Z	16 DeM
	18	~ Z	17 &E

6. Validity

a. Derive: ~ B

1	(R ⊃ C) ∨ (B ⊃ C)	Assumption
2	~ (E & A) ⊃ ~ (R ⊃ C)	Assumption
3	~ E & ~ C	Assumption
4	~ E	3 &E
5	~ E ∨ ~ A	4 ∨I
6	~ (E & A)	5 DeM
7	~ (R ⊃ C)	2, 6 ⊃E
8	B ⊃ C	1, 7 DS
9	~ C	3 &E
10	~ B	8, 9 MT

c. Derive: ~ W ⊃ ~ A

```
 1 | A ⊃ [W ∨ ~ (C ∨ R)]            Assumption
 2 | ~ R ⊃ C                        Assumption
   |--------------------
 3 | | ~ W                          A / ⊃I
   | |------------------
 4 | | | A                          A / ~ I
   | | |----------------
 5 | | | W ∨ ~ (C ∨ R)              1, 4 ⊃E
 6 | | | ~ (C ∨ R)                  3, 5 DS
 7 | | | ~ ~ R ∨ C                  2 Impl
 8 | | | R ∨ C                      7 DN
 9 | | | C ∨ R                      8 Com
10 | | ~ A                          4–9 ~ I
11 | ~ W ⊃ ~ A                      3–10 ⊃I
```

e. Derive: J ⊃ ~ (E ∨ ~ M)

```
 1 | ~ (J & ~ H)                    Assumption
 2 | ~ H ∨ M                        Assumption
 3 | E ⊃ ~ M                        Assumption
   |----------------
 4 | | J                            A / ⊃I
   | |--------------
 5 | | ~ J ∨ ~ ~ H                  1 DeM
 6 | | ~ ~ J                        4 DN
 7 | | ~ ~ H                        5, 6 DS
 8 | | M                            2, 7 DS
 9 | | ~ ~ M                        8 DN
10 | | ~ E                          3, 9 MT
11 | | ~ E & ~ ~ M                  10, 9 &I
12 | | ~ (E ∨ ~ M)                  11 DeM
13 | J ⊃ ~ (E ∨ ~ M)                4–12 ⊃I
```

g. Derive: ~ A ⊃ [H ⊃ (F & B)]

```
 1 | (H & ~ S) ⊃ A                  Assumption
 2 | ~ B ⊃ ~ S                      Assumption
 3 | ~ S ∨ C                        Assumption
 4 | C ⊃ F                          Assumption
   |--------------------
 5 | | ~ A                          A / ⊃I
   | |------------------
 6 | | | H                          A / ⊃I
   | | |----------------
 7 | | | H ⊃ (~ S ⊃ A)              1 Exp
 8 | | | ~ S ⊃ A                    6, 7 ⊃E
 9 | | | ~ ~ S                      5, 8 MT
10 | | | C                          3, 9 DS
11 | | | F                          4, 10 ⊃E
12 | | | ~ ~ B                      2, 9 MT
13 | | | B                          12 DN
14 | | | F & B                      11, 13 &I
15 | | H ⊃ (F & B)                  6–14 ⊃I
16 | ~ A ⊃ [H ⊃ (F & B)]            5–15 ⊃I
```

7. Inconsistency

a.	1	B ∨ ~ C	Assumption
	2	(L ⊃ ~ G) ⊃ C	Assumption
	3	(G ≡ ~ B) & (~ L ⊃ ~ B)	Assumption
	4	~ L	Assumption
	5	~ L ∨ ~ G	4 ∨I
	6	L ⊃ ~ G	5 Impl
	7	C	2, 6 ⊃E
	8	~ L ⊃ ~ B	3 &E
	9	~ B	4, 8 ⊃E
	10	~ C	1, 9 DS

8.a. The rules of replacement are two-way rules. If we can derive **Q** from **P** by using only these rules, we can derive **P** from **Q** by using the rules in reverse order.

c. Suppose that before a current line **n** of a derivation, an accessible line **i** contains a sentence of the form **P** ⊃ **Q**. The sentence **P** ⊃ (**P** & **Q**) can be derived by using the following routine:

i	**P** ⊃ **Q**	
n	**P**	Assumption
n + 1	**Q**	**i**, **n** ⊃E
n + 2	**P** & **Q**	**n**, **n** + 1 &E
n + 3	**P** ⊃ (**P** & **Q**)	**n** − **n** + 2 ⊃I

ERRORS IN THE “STUDENT SOLUTIONS MANUAL” OF
THE LOGIC BOOK
(CHAPTER 5)

p. 1

(i) For the solution to exercise 5.1.1E(c), read

2-6 ⊃I

instead of

1-6 ⊃I

on line 7 of the derivation.

(ii) For the solution to exercise 5.1.1E(e), read

3, 4 ⊃E

instead of

3, 6 ⊃E

on line 7 of the derivation.

p. 5

The solutions to the unstarred exercises in Section 5.1.5E are missing altogether from the “Student Solutions Manual.” They are as follows:

1a) line 1: Assumption
line 2: 1, &E
line 3: 1, &E
line 4: A / ∨E
line 5: 3, 4 &I
line 6: 5, ∨I
line 7: A / ∨E
line 8: 3, 7 &I
line 9: 8, ∨I
line 10: 2, 4-6, 7-9 ∨E

1c) line 1: Assumption
line 2: A / ~I
line 3: 1, 2 ⊃E
line 4: 3 &E
line 5: 2R
line 6: 2-5 ~I

1e) line 1: Assumption
line 2: Assumption
line 3: Assumption
line 4: Assumption
line 5: Assumption
line 6: A / I⊃
line 7: 2, 6 ⊃E
line 8: 4, 7 ⊃E
line 9: 8 &E
line 10: 9 vI
line 11: 3, 10 ⊃E
line 12: A / ~E
line 13: 9R
line 14: 1R
line 15: 12-14 ~E
line 16: 11, 15 &I
line 17: 6-16 I⊃

1g) line 1: Assumption
line 2: A / ≡I
line 3: A / ~E
line 4: 1 &E
line 5: 2R
line 6: 3-5 ~E
line 7: A / ≡I
line 8: A / ~E
line 9: 7R
line 10: 1 &E
line 11: 8-10 ~E
line 12: 2-6, 7-11 ≡I

2a) The mistake occurs at line 3. '~A' is not the antecedent of '~~A ⊃ (B & ~D)'.

2c) The mistake occurs at line 7. The 'A' on line 4 of the derivation is not acessible at line 7 of the derivation.

p. 11

In the first derivation on p. 11 of Chap. 5 of the "Student Solutions Manual," read

5-11 ~E

instead of

5-11 ~I

on line 12 of the derivation.

p. 12

In the first derivation on p. 12 of Chap. 5 of the "Student Solutions Manual" the number '11' is skipped in the numbering of the lines. The line labeled '12' should be labeled '11', the line labeled '13' should be labeled '12', etc.

p. 16

In the second derivation on p. 16 of Chap. 5 of the "Student Solutions Manual," read

2, 3 ⊃E

instead of

2, 3 ⊃I

on line 4 of the derivation.

p. 18

In the third derivation on p. 18 of Chap. 5 of the "Student Solutions Manual," read

1, 2-7, 8-13 ∨E

instead of

1, 2-6, 7-13 ∨E

on line 14 of the derivation.

p. 19

(i) In the first derivation on p. 19 of Chap. 5 of the "Student Solutions Manual," read

A / ~E

instead of

A / ~I

on line 2 of the derivation.

(ii) In the same derivation, read

A / ≡I

instead of

A / ⊃I

on line 9 of the derivation.

p. 27

In the derivation on p. 27 of Chap. 5 of the "Student Solutions Manual," the numbering of lines on the left skips '18' and '19'. Read '18' instead of '20'.

Section 6.1E

1.a. We shall prove that every sentence of *SL* that contains only binary connectives, if any, is true on every truth-value assignment on which all its atomic components are true. Hence every sentence of *SL* that contains only binary connectives is true on at least one truth-value assignment, and thus no such sentence can be truth-functionally false. We proceed by mathematical induction on the number of occurrences of connectives in such sentences. (Note that we need not consider *all* sentences of *SL* in our induction but only those with which the thesis is concerned.)

Basis clause: Every sentence with zero occurrences of a binary connective (and no occurrences of unary connectives) is true on every truth-value assignment on which all its atomic components are true.

Inductive step: If every sentence with **k** or fewer occurrences of binary connectives (and no occurrences of unary connectives) is true on every truth-value assignment on which all its atomic components are true, then every sentence with **k** + 1 occurrences of binary connectives (and no occurrences of unary connectives) is true on every truth-value assignment on which all its atomic components are true.

The proof of the basis clause is straightforward. A sentence with zero occurrences of a connective is an atomic sentence, and each atomic sentence is true on every truth-value assignment on which its atomic component (which is the sentence itself) is true.

The inductive step is also straightforward. Assume that the thesis holds for every sentence of *SL* with **k** or fewer occurrences of binary connectives and no unary connectives. Any sentence **P** with **k** + 1 occurrences of binary connectives and no unary connectives must be of one of the four forms **Q** & **R**, **Q** ∨ **R**, **Q** ⊃ **R**, and **Q** ≡ **R**. In each case **Q** and **R** contain **k** or fewer occurrences of binary connectives, so the inductive hypothesis holds for both **Q** and **R**. That is, both **Q** and **R** are true on every truth-value assignment on which all their atomic components are true. Since **P**'s immediate components are **Q** and **R**, its atomic components are just those of **Q** and **R**. But conjunctions, disjunctions, conditionals, and biconditionals are true when both their immediate components are true. So **P** is also true on every truth-value assignment on which its atomic components are true, for both its immediate components are then true. This completes our proof. (Note that in this clause we ignored sentences of the form ~ **Q**, for the thesis concerns only those sentences of *SL* that contain *no* occurrences of '~'.)

b. Every sentence **P** that contains no binary connectives either contains no connectives or contains at least one occurrence of '~'. We prove the thesis by mathematical induction on the number of occurrences of '~' in such

sentences. The first case consists of the atomic sentences of *SL* since these contain zero occurrences of connectives.
Basis clause: Every atomic sentence is truth-functionally indeterminate.
Inductive step: If every sentence with **k** or fewer occurrences of '~' (and no binary connectives) is truth-functionally indeterminate, then every sentence with **k** + 1 occurrences of '~' (and no binary connectives) is truth-functionally indeterminate.

The basis clause is obvious.

The inductive step is also obvious. Suppose **P** contains **k** + 1 occurrences of '~' and no binary connectives and that the thesis holds for every sentence with fewer than **k** + 1 occurrences of '~' and no binary connectives. **P** is a sentence of the form ~ **Q**, where **Q** contains **k** occurrences of '~'; hence, by the inductive hypothesis, **Q** is truth-functionally indeterminate. The negation of a truth-functionally indeterminate sentence is also truth-functionally indeterminate. Hence ~ **Q**, that is, **P**, is truth-functionally indeterminate. This completes the induction.

c. The induction is on the number of occurrences of connectives in **P**. The thesis to be proved is

> If two truth-value assignments **A**′ and **A**″ assign the same truth-values to the atomic components of a sentence **P**, then **P** has the same truth-value on **A**′ and **A**″.

Basis clause: The thesis holds for every sentence with zero occurrences of connectives.
Inductive step: If the thesis holds for every sentence with **k** or fewer occurrences of connectives, then the thesis holds for every sentence with **k** + 1 occurrences of connectives.

The basis clause is obvious. If **P** contains zero occurrences of connectives, then **P** is an atomic sentence and its own only atomic component. **P** must have the same truth-value on **A**′ and **A**″ because *ex hypothesi* it is assigned the same truth-value on each assignment.

To prove the inductive step, we let **P** be a sentence with **k** + 1 occurrences of connectives and assume that the thesis holds for every sentence containing **k** or fewer occurrences of connectives. Then **P** is of the form ~ **Q**, **Q** & **R**, **Q** ∨ **R**, **Q** ⊃ **R**, or **Q** ≡ **R**. In each case the immediate component(s) of **P** contain **k** or fewer occurrences of connectives and hence fall under the inductive hypothesis. So each immediate component of **P** has the same truth-value on **A**′ and **A**″. **P** therefore has the same truth-value on **A**′ and **A**″, as determined by the characteristic truth-tables.

d. We prove the thesis by mathematical induction on the number of conjuncts in an iterated conjunction of sentences $\mathbf{P}_1$, . . . , $\mathbf{P}_\mathbf{n}$ of *SL*.
Basis clause: Every iterated conjunction of just one sentence of *SL* is true on a truth-value assignment if and only if that one sentence is true on that assignment.
Inductive step: If every iterated conjunction of **k** or fewer sentences of *SL* is true

on a truth-value assignment if and only if each of those conjuncts is true on that assignment, then every iterated conjunction of **k** + 1 sentences of *SL* is true on a truth-value assignment if and only if each of those conjuncts is true on that assignment.

The basis clause is trivial.

To prove the inductive step, we assume that the thesis holds for iterated conjunctions of **k** or fewer sentences of *SL*. Let **P** be an iterated conjunction of **k** + 1 sentences. Then **P** is **Q** & **R**, where **Q** is an iterated conjunction of **k** sentences. **P** is therefore an iterated conjunction of all the sentences of which **Q** is an iterated conjunction, and **R**. By the inductive hypothesis, the thesis holds of **Q**; that is, **Q** is true on a truth-value assignment if and only if the sentences of which **Q** is an iterated conjunction are true on that assignment. Hence, whenever all the sentences of which **P** is an iterated conjunction are true, both **Q** and **R** are true, and thus **P** is true as well. Whenever at least one of those sentences is false, either **Q** is false or **R** is false, making **P** false as well. Hence **P** is true on a truth-value assignment if and only if all the sentences of which it is an iterated conjunction are true on that assignment.

e. We proceed by mathematical induction on the number of occurrences of connectives in **P**. The argument is

> The thesis holds for every atomic sentence **P**.
>
> If the thesis holds for every sentence **P** with **k** or fewer occurrences of connectives, then it holds for every sentence **P** with **k** + 1 occurrences of connectives.
>
> ---
>
> The thesis holds for every sentence **P** of *SL*.

The proof of the basis clause is fairly simple. If **P** is an atomic sentence and **Q** is a sentential component of **P**, then **Q** must be identical with **P** (since each atomic sentence is its own only atomic component). For any sentence $\mathbf{Q}_1$, then, $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is simply the sentence $\mathbf{Q}_1$. Here it is trivial that if **Q** and $\mathbf{Q}_1$ are truth-functionally equivalent, so are **P** (which is just **Q**) and $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ (which is just $\mathbf{Q}_1$).

In proving the inductive step, the following result will be useful:

6.1.1. If **Q** and $\mathbf{Q}_1$ are truth-functionally equivalent and **R** and $\mathbf{R}_1$ are truth-functionally equivalent, then each of the following pairs are pairs of truth-functionally equivalent sentences:

$\sim \mathbf{Q}$	$\sim \mathbf{Q}_1$
$\mathbf{Q} \,\&\, \mathbf{R}$	$\mathbf{Q}_1 \,\&\, \mathbf{R}_1$
$\mathbf{Q} \vee \mathbf{R}$	$\mathbf{Q}_1 \vee \mathbf{R}_1$
$\mathbf{Q} \supset \mathbf{R}$	$\mathbf{Q}_1 \supset \mathbf{R}_1$
$\mathbf{Q} \equiv \mathbf{R}$	$\mathbf{Q}_1 \equiv \mathbf{R}_1$

Proof: The truth-value of a molecular sentence is wholly determined by the truth-values of its immediate components. Hence, if there is a truth-value assignment on which some sentence in the left-hand column has a truth-value different from that of its partner in the right-hand column, then on that assignment either **Q** and $\mathbf{Q}_1$ have different truth-values or **R** and $\mathbf{R}_1$ have different truth-values. But this is impossible because *ex hypothesi* **Q** and $\mathbf{Q}_1$ are truth-functionally equivalent and **R** and $\mathbf{R}_1$ are truth-functionally equivalent.

To prove the inductive step of the thesis, we assume the inductive hypothesis: that the thesis holds for every sentence with **k** or fewer occurrences of connectives. Let **P** be a sentence of *SL* with **k** + 1 occurrences of connectives, let **Q** be a sentential component of **P**, let $\mathbf{Q}_1$ be a sentence that is truth-functionally equivalent to **Q**, and let $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ be a sentence that results from replacing one or more occurrences of **Q** in **P** with $\mathbf{Q}_1$. Suppose, first, that **Q** is identical with **P**. Then, by the reasoning in the proof of the basis clause, it follows trivially that **P** and $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ are truth-functionally equivalent. Now suppose that **Q** is a sentential component of **P** that is *not* identical with **P** (in which case we say that **Q** is a *proper* sentential component of **P**). Either **P** is of the form ~ **R** or **P** has a binary connective as its main connective and is of one of the four forms **R** & **S**, **R** ∨ **S**, **R** ⊃ **S**, and **R** ≡ **S**. We shall consider the two cases separately.

i. **P** is of the form ~ **R**. Since **Q** is a proper sentential component of **P**, **Q** must be a sentential component of **R**. Hence $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is a sentence ~ $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$. But **R** has **k** occurrences of connectives, so by the inductive hypothesis, **R** is truth-functionally equivalent to $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$. It follows from 6.1.1 that ~ **R** is truth-functionally equivalent to ~ $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$; that is, **P** is truth-functionally equivalent to $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$.

ii. **P** is of the form **R** & **S**, **R** ∨ **S**, **R** ⊃ **S**, or **R** ≡ **S**. Since **Q** is a proper component of **P**, $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ must be **P** with its left immediate component replaced by a sentence $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$, **P** with its right immediate component replaced with a sentence $[\mathbf{S}](\mathbf{Q}_1//\mathbf{Q})$, or **P** with both replacements made. Both **R** and **S** have fewer than **k** + 1 occurrences of connectives, and so the inductive hypothesis holds for both **R** and **S**. Hence **R** is truth-functionally equivalent to $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$, and **S** is truth-functionally equivalent to $[\mathbf{S}](\mathbf{Q}_1//\mathbf{Q})$. And **R** is truth-functionally equivalent to **R** and **S** is truth-functionally equivalent to **S**. Whatever replacements are made in **P**, it follows by 6.1.1 that **P** is truth-functionally equivalent to $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$.

This completes the proof of the inductive step and thus the proof of our thesis.

2. An example of a sentence that contains only binary connectives and is truth-functionally true is ‘A ⊃ A’. An attempted proof would break down in the proof of the inductive step (since no atomic sentence is truth-functionally true, the basis clause will go through).

Section 6.2E

1. Suppose that we have constructed, in accordance with the algorithm, a sentence for a row of a truth-function schema that defines a truth-function of **n** arguments. We proved in Exercise 1.d in Section 6.1E the result that an iterated conjunction (. . . ($\mathbf{P}_1$ & $\mathbf{P}_2$) & . . . & $\mathbf{P}_\mathbf{n}$) is true on a truth-value assignment if and only if $\mathbf{P}_1$, . . . , $\mathbf{P}_\mathbf{n}$ are all true on that truth-value assignment. We have constructed the present iterated conjunction of atomic sentences and negations of atomic sentences in such a way that each conjunct is true when the atomic components have the truth-values represented in that row. Hence for that assignment the sentence constructed is true. For any other assignments to the atomic components of the sentence, at least one of the conjuncts is false; hence the conjunction is also false.

2.a. (A & ~ B) ∨ (~ A & ~ B)
b. A & ~ A
d. ([(A & B) & C] ∨ [(A & B) & ~ C]) ∨ [(~ A & ~ B) & C]

3. Suppose that the table defines a truth-function of **n** arguments. We first construct an iterated disjunction of **n** disjuncts such that the **i**th disjunct is the negation of the **i**th atomic sentence of *SL* if the **i**th truth-value in the row is **T**, and the **i**th disjunct is the **i**th atomic sentence of *SL* if the **i**th truth-value in the row is **F**. Note that this iterated disjunction is *false* exactly when its atomic components have the truth-values displayed in that row. We then negate the iterated disjunction, to obtain a sentence that is *true* for those truth-values and false for all other truth-values that may be assigned to its atomic components.

4. To prove that {'~', '&'} is truth-functionally complete, it will suffice to show that for each sentence of *SL* containing only '~', '∨', and '&', there is a truth-functionally equivalent sentence of *SL* that contains the same atomic components and in which the only connectives are '~' and '&'. For it will then follow, from the fact that {'~', '∨', '&'} is truth-functionally complete, that {'~', '&'} is also truth-functionally complete. But every sentence of the form

P ∨ **Q**

is truth-functionally equivalent to

~ (~ **P** & ~ **Q**)

So by repeated substitutions, we can obtain, from sentences containing '~', '∨', and '&', truth-functionally equivalent sentences that contain only '~' and '&'.

To show that {'~', '⊃'} is truth-functionally complete, it suffices to point out that every sentence of the form

P & **Q**

is truth-functionally equivalent to the corresponding sentence

$$\sim (\mathbf{P} \supset \sim \mathbf{Q})$$

and that every sentence of the form

$$\mathbf{P} \vee \mathbf{Q}$$

is truth-functionally equivalent to the corresponding sentence

$$\sim \mathbf{P} \supset \mathbf{Q}$$

For then we can find, for each sentence containing only '~', '∨', and '&', a truth-functionally equivalent sentence with the same atomic components containing only '~' and '⊃'. It follows that {'~', '⊃'} is truth-functionally complete, since {'~', '∨', '&'} is.

5. To show this, we need only note that the negation and disjunction truth-functions can be expressed using only the dagger. The truth-table for 'A ↓ A' is

A	A	↓	A
T	**T**	**F**	**T**
F	**F**	**T**	**F**

The sentence 'A ↓ A' expresses the negation truth-function, for the column under the dagger is identical with the column to the right of the vertical line in the characteristic truth-table for negation.

The disjunction truth-function is expressed by '(A ↓ B) ↓ (A ↓ B)', as the following truth-table shows:

A	B	(A	↓	B)	↓	(A	↓	B)
T	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**F**
F	**T**	**F**	**F**	**T**	**T**	**F**	**F**	**T**
F	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**

This table shows that '(A ↓ B) ↓ (A ↓ B)' is true on every truth-value assignment on which at least one of 'A' and 'B' is true. Hence that sentence expresses the disjunction truth-function.

Thus any truth-function that is expressed by a sentence of *SL* containing only the connectives '~' and '∨' can be expressed by a sentence containing only '↓' as a connective. To form such a sentence, we convert the sentence of *SL* containing just '~' and '∨' that expresses the truth-function in question as follows. Repeatedly replace components of the form ~ **P** with **P** ↓ **P** and components of the form $\mathbf{P} \vee \mathbf{Q}$ with $(\mathbf{P} \downarrow \mathbf{Q}) \downarrow (\mathbf{P} \downarrow \mathbf{Q})$ until a sentence containing ‘↓’ as the only connective is obtained. Since {‘∨’, ‘~’} is truth-functionally complete, so is {‘↓’}.

7. The set {‘~’} is not truth-functionally complete because every sentence containing only ‘~’ is truth-functionally indeterminate. Hence truth-functions expressed in *SL* by truth-functionally true sentences and truth-functions expressed in *SL* truth-functionally false sentences cannot be expressed by a sentence that contains only ‘~’.

The set {‘&’, ‘∨’, ‘⊃’, ‘≡’} is not truth-functionally complete because no sentence that contains only binary connectives (if any) is truth-functionally false. Hence no truth-function that is expressed in *SL* by a truth-functionally false sentence can be expressed by a sentence containing only binary connectives of *SL*.

8. We shall prove by mathematical induction that in the truth-table for a sentence **P** containing only the connectives ‘~’ and ‘≡’ and two atomic components, the column under the main connective of **P** has an even number of **T**s and an even number of **F**s. For then we shall know that no sentence containing only those connectives can express, for example, the truth-function defined as follows (the material conditional truth-function):

T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**T**

In the induction remember that any sentence of *SL* that contains two atomic components has a four-row truth-table. Our induction will proceed on the number of occurrences of connectives in **P**. However, the first case, that considered in the basis clause, is the case where **P** contains *one* occurrence of a connective. This is because every sentence that contains zero occurrences of connectives is an atomic sentence and thus cannot contain more than one atomic component.

Basis clause: The thesis holds for every sentence of *SL* with exactly two atomic components and one occurrence of (one of) the connectives ‘~’ and ‘≡’.

In this case **P** cannot be of the form $\sim \mathbf{Q}$, for if the initial ‘~’ is the only connective in **P**, then **Q** is atomic, and hence **P** does not contain two atomic components. So **P** is of the form $\mathbf{Q} \equiv \mathbf{R}$, where **Q** and **R** are atomic sentences. $\mathbf{Q} \equiv \mathbf{R}$ will have to be true on assignments that assign the same truth-values to **Q** and **R** and false on other assignments. Hence the thesis holds in this case.

Inductive step: If the thesis holds for every sentence of *SL* that contains **k** or fewer occurrences of the connectives ‘~’ and ‘≡’ (and no other connectives) and two atomic components, then the thesis holds for every sentence of *SL* that contains two atomic components and **k** + 1 occurrences of the connectives '~' and '≡' (and no other connectives).

Let **P** be a sentence of *SL* that contains exactly two atomic components and **k** + 1 occurrences of the connectives '~' and '≡' (and no other connectives). There are two cases to consider.

i. **P** is of the form ~ **Q**. Then **Q** falls under the inductive hypothesis; hence in the truth-table for **Q** the column under the main connective contains an even number of **T**s and an even number of **F**s. The column for the sentence ~ **Q** simply reverses the **T**s and **F**s, so it also contains an even number of **T**s and an even number of **F**s.

ii. **P** is of the form **Q** ≡ **R**. Then **Q** and **R** each contain fewer occurrences of connectives. If, in addition, **Q** and **R** each contain both of the atomic components of **P**, then they fall under the inductive hypothesis—**Q** has an even number of **T**s and an even number of **F**s in its truth-table column, and so does **R**. On the other hand, if **Q** or **R** (or both) only contains one of the atomic components of **P** (e.g., if **P** is '~ A ≡ (B ≡ A)' then **Q** is '~ A'), then **Q** or **R** (or both) fails to fall under the inductive hypothesis. However, in this case the component in question also has an even number of **T**s and an even number of **F**s in its column in the truth-table for **P**. This is because (a) two rows assign **T** to the single atomic component of **Q** and, by the result in Exercise 1.c, **Q** has the same truth-value in these two rows; and (b) two rows assign **F** to the single atomic component of **Q** and so, by the same result, **Q** has the same truth-value in these two rows.

We will now show that if **Q** and **R** each have an even number of **T**s and an even number of **F**s in their truth-table columns, then so must **P**. Let us assume the contrary, that is, we shall suppose that **P** has an odd number of **T**s and an odd number of **F**s in its truth-table column. There are then two possibilities.

a. There are 3 **T**s and 1 **F** in **P**'s truth-table column. Then in three rows of their truth-table columns, **Q** and **R** have the same truth-value, and in one row they have different truth-values. So either **Q** has one more **T** in its truth-table column than does **R**, or vice-versa. Either way, since the sum of an even number plus 1 is odd, it follows that either **Q** has an odd number of **T**s in its truth-table column or **R** has an odd number of **T**s in its truth-table column. This contradicts our inductive hypothesis, so we conclude that **P** cannot have 3 **T**s and 1 **F** in its truth-table column.

b. There are 3 **F**s and 1 **T** in **P**'s truth-table column. By reasoning similar to that just given, it is easily shown that this is impossible, given the inductive hypothesis.

Therefore **P** must have an even number of **T**s and **F**s in its truth-table column.

9. First, a binary connective whose unit set is truth-functionally complete must be such that a sentence of which it is the main connective is false whenever all its immediate components are true. Otherwise, every sentence containing only that connective would be true whenever its atomic components were. And then, for example, the negation truth-function would not be expressible using that connective. Similar reasoning shows that the main column of the characteristic truth-table must contain **T** in the last row. Otherwise, no sentence containing that connective could be truth-functionally true.

Second, the column in the characteristic truth-table must contain an odd number of **T**s and an odd number of **F**s. For otherwise, as the induction in Exercise 8 shows, any sentence containing two atomic components and only this connective would have an even number of **T**s and an even number of **F**s in its truth-table column. The disjunction truth-function, for example, would then not be expressible.

Combining these two results, it is easily verified that there are only two possible characteristic truth-tables for a binary connective whose unit set is truth-functionally complete—that for '$\downarrow$' and that for '$|$'.

Section 6.3E

1.a. $\{A \supset B, C \supset D\}$, $\{A \supset B\}$, $\{C \supset D\}$, $\varnothing$

b. $\{C \vee \sim D, \sim D \vee C, C \vee C\}$, $\{C \vee \sim D, \sim D \vee C\}$, $\{C \vee \sim D, C \vee C\}$, $\{\sim D \vee C, C \vee C\}$, $\{C \vee \sim D\}$, $\{\sim D \vee C\}$, $\{C \vee C\}$, $\varnothing$

c. $\{(B \,\&\, A) \equiv K\}$, $\varnothing$

d. $\varnothing$

2.a, b, d, e.

4.a. To prove that *SD** is sound, it suffices to add a clause for the new rule to the induction in the proof of Metatheorem 6.3.1.

13. If $\mathbf{Q}_{\mathbf{k}+1}$ at position $\mathbf{k} + 1$ is justified by $\sim \equiv$I, then $\mathbf{Q}_{\mathbf{k}+1}$ is a negated biconditional.

$$\begin{array}{r|ll} \mathbf{h} & \mathbf{P} & \\ \mathbf{j} & \sim \mathbf{Q} & \\ \mathbf{k} + 1 & \sim (\mathbf{P} \equiv \mathbf{Q}) & \quad \mathbf{h}, \mathbf{j} \ \sim \equiv \mathrm{I} \end{array}$$

By the inductive hypothesis, $\Gamma_{\mathbf{h}} \vDash \mathbf{P}$ and $\Gamma_{\mathbf{j}} \vDash \sim \mathbf{Q}$. Since $\mathbf{P}$ and $\sim \mathbf{Q}$ are accessible at position $\mathbf{k} + 1$, every member of $\Gamma_{\mathbf{h}}$ is a member of $\Gamma_{\mathbf{k}+1}$, and every member of $\Gamma_{\mathbf{j}}$ is a member of $\Gamma_{\mathbf{k}+1}$. Hence, by 6.3.2, $\Gamma_{\mathbf{k}+1} \vDash \mathbf{P}$ and $\Gamma_{\mathbf{k}+1} \vDash \sim \mathbf{Q}$. But $\sim (\mathbf{P} \equiv \mathbf{Q})$ is true whenever $\mathbf{P}$ and $\sim \mathbf{Q}$ are both true. So $\Gamma_{\mathbf{k}+1} \vDash \sim (\mathbf{P} \equiv \mathbf{Q})$ as well.

c. To show that *SD** is not sound, it suffices to give an example of a derivation in *SD** of a sentence $\mathbf{P}$ from a set Γ of sentences such that $\mathbf{P}$ is *not* truth-functionally entailed by Γ. That is, we show that for some Γ and $\mathbf{P}$,

$\Gamma \vdash \mathbf{P}$ in *SD**, but $\Gamma \nvDash \mathbf{P}$. Here is an example:

1	A	Assumption
2	A ∨ B	Assumption
3	B	1, 2 C∨E

It is easily verified that {A, A ∨ B} does not truth-functionally entail 'B'.

e. Yes. In proving Metatheorem 6.3.1, we showed that each rule of *SD* is truth-preserving. It follows that if every rule of *SD** is a rule of *SD,* then every rule of *SD** is truth-preserving. Of course, as we saw in Exercise 4.c, *adding* a rule produces a system that is not sound if the rule is not truth-preserving.

5. No. In *SD* we can derive **Q** from a sentence **P** & **Q** by &E. But, if '&' had the suggested truth-table, then {**P** & **Q**} would *not* truth-functionally entail **Q**, for (by the second row of the table) **P** & **Q** would be true when **P** is true and **Q** is false. Hence it would be the case that {**P** & **Q**} ⊢ **Q** in *SD* but not the case that {**P** & **Q**} ⊨ **Q**.

6. To prove that *SD+* is sound for sentential logic, we must show that the rules of *SD+* that are not rules of *SD* are truth-preserving. (By Metatheorem 6.3.1, the rules of *SD* have been shown to be truth-preserving.) The three additional rules of inference in *SD+* are Modus Tollens, Hypothetical Syllogism, and Disjunctive Syllogism. We introduced each of these rules in Chapter 5 as a *derived* rule. For example, we showed that Modus Tollens is eliminable, that anything that can be derived using this rule can be derived without it, using just the smaller set of rules in *SD.* It follows that each of these three rules is truth-preserving. For if use of one of these rules can lead from true sentences to false ones, then we can construct a derivation in *SD* (without using the derived rule) in which the sentence derived is not truth-functionally entailed by the set consisting of the undischarged assumptions. But Metatheorem 6.3.1 shows that this is impossible. Hence each of the derived rules is truth-preserving.

All that remains to be shown, in proving that *SD+* is sound, is that the rules of replacement are also truth-preserving. We can incorporate this as a thirteenth case in the proof of the inductive step for Metatheorem 6.3.1:

13. If $\mathbf{Q}_{\mathbf{k}+1}$ at position **k** + 1 is justified by a rule of replacement, then $\mathbf{Q}_{\mathbf{k}+1}$ is derived as follows:

h	**P**	
k + 1	[**P**]($\mathbf{Q}_1$//**Q**)	**h** RR

where RR is some rule of replacement, sentence **P** at position **h** is accessible at position **k** + 1, and [**P**]($\mathbf{Q}_1$//**Q**) is a sentence that is the result of replacing a component **Q** of **P** with a component $\mathbf{Q}_1$ in accordance with one of the rules of replacement. That the sentence **Q** is truth-functionally equivalent to $\mathbf{Q}_1$, no matter what the rule of replacement is, is easily verified. So, by Exercise 1.e in Section 6.1**E**, $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is truth-functionally equivalent to **P**. By the inductive hypothesis, $\Gamma_\mathbf{k} \vDash \mathbf{P}$; and since **P** at **h** is accessible at position **k** + 1, it follows that $\Gamma_{\mathbf{k}+1} \vDash \mathbf{P}$. But $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is true whenever **P** is true (since they are truth-functionally equivalent), so $\Gamma_{\mathbf{k}+1} \vDash [\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$; that is, $\Gamma_{\mathbf{k}+1} \vDash \mathbf{Q}_{\mathbf{k}+1}$.

Section 6.4E

1. Proof of 6.4.4 Assume that $\Gamma \vdash \mathbf{P}$ in *SD*. Then there is a derivation in *SD* of the following sort

$$\begin{array}{r|l}
1 & \mathbf{P}_1 \\
. & . \\
\mathbf{n} & \mathbf{P}_\mathbf{n} \\
\cline{2-2}
. & . \\
\mathbf{m} & \mathbf{P}
\end{array}$$

(where $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_\mathbf{n}$ are members of Γ). To show that $\Gamma \cup \{\sim \mathbf{P}\}$ is inconsistent in *SD*, we need only produce a derivation of some sentence **Q** and ~ **Q** from members of $\Gamma \cup \{\sim \mathbf{P}\}$. This is easy. Start with the derivation of **P** from Γ and add ~ **P** as a new primary assumption at line **n** + 1, renumbering subsequent lines as is appropriate. As a new last line, enter ~ **P** by Reiteration. The result is a derivation of the sort

$$\begin{array}{r|ll}
1 & \mathbf{P}_1 & \\
. & . & \\
\mathbf{n} & \mathbf{P}_\mathbf{n} & \\
\mathbf{n} + 1 & \sim \mathbf{P} & \\
\cline{2-2}
. & . & \\
\mathbf{m} + 1 & \mathbf{P} & \\
\mathbf{m} + 2 & \sim \mathbf{P} & \quad \mathbf{n} + 1\ \mathbf{R}
\end{array}$$

So if $\Gamma \vdash \mathbf{P}$, then $\Gamma \cup \{\sim \mathbf{P}\}$ is inconsistent in *SD*.

Now assume that $\Gamma \cup \{\sim \mathbf{P}\}$ is inconsistent in *SD*. Then there is a derivation in *SD* of the sort

$$\begin{array}{r|l}
1 & \mathbf{P}_1 \\
. & . \\
\mathbf{n} & \mathbf{P}_\mathbf{n} \\
\mathbf{n} + 1 & \sim \mathbf{P} \\
\cline{2-2}
. & . \\
\mathbf{m} & \mathbf{Q} \\
. & . \\
\mathbf{p} & \sim \mathbf{Q}
\end{array}$$

(where $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P_n}$ all members of Γ). To show that $\Gamma \vdash \mathbf{P}$, we need only produce a derivation in which the primary assumptions are members of Γ and the last line is **P**. This is easy. Start with this derivation, but make ~ **P** an auxiliary assumption rather than a primary assumption. Enter **P** as a new last line, justified by Negation Elimination. The result is a derivation of the sort

```
    1 | P1
    . | .
    n | Pn
      |--------
n + 1 |  | ~ P
      |  |-----
    . |  | .
    m |  | Q
    . |  | .
    p |  | ~ Q
p + 1 | P            n + 1 − p   ~ E
```

Proof of 6.4.10. Assume $\Gamma \cup \{\mathbf{P}\}$ is inconsistent in *SD*. Then there is a derivation in *SD* of the sort

```
    1 | P1
    . |
    n | Pn
n + 1 | P
      |-----
    . | .
    m | Q
    . | .
    p | ~ Q
```

(where $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P_n}$ are members of Γ). But then there is also a derivation of the following sort

```
    1 | P1
    . | .
    n | Pn
      |--------
n + 1 |  | P
      |  |-----
      |  | .
    m |  | Q
    p |  | ~ Q
p + 1 | ~ P          n + 1 − p   ~ I
```

This shows that if $\Gamma \cup \{\mathbf{P}\}$ is inconsistent in *SD*, then $\Gamma \vdash$ ~ **P** in *SD*.

2. If Γ is inconsistent in *SD* then, by the definition of inconsistency in *SD*, there is some sentence **P** such that both **P** and ~ **P** are derivable in *SD* from Γ. By the definition of derivability in *SD*, there is a derivation in which all of the primary assumptions are members of Γ and **P** occurs in the scope of only those assumptions, and there is a derivation in which all of the primary assumptions are members of Γ and ~ **P** occurs in the scope of only those assumptions. Because all derivations are finite in length, it follows that only a finite subset of members of Γ occurs as primary assumptions in each of these derivations, i.e., **P** is derivable from a finite subset Γ' of Γ and ~ **P** is derivable from a finite subset Γ'' of Γ. We can extend the derivation of **P** from Γ' to a derivation of **P** from $\Gamma' \cup \Gamma''$ by adding members of Γ'' that are not members of Γ' as primary assumptions in that derivation, and we can extend the derivation of ~ **P** from Γ'' to a derivation of ~ **P** from $\Gamma' \cup \Gamma''$ by adding members of Γ' that are not members of Γ'' as primary assumptions in that derivation. This establishes that both **P** and ~ **P** are derivable from the finite subset $\Gamma' \cup \Gamma''$ of Γ, and hence that there is a finite subset of Γ that is inconsistent in *SD*.

4. Since every rule of *SD* is a rule of *SD+*, every derivation in *SD* is a derivation in *SD+*. So if $\Gamma \vDash \mathbf{P}$, then $\Gamma \vdash \mathbf{P}$ in *SD*, by Metatheorem 6.4.1, and therefore $\Gamma \vdash \mathbf{P}$ in *SD+*. That is, *SD+* is complete for sentential logic.

7. a. Since we already know that *SD* is complete, we need only show that wherever Reiteration is used in a derivation in *SD*, it can be eliminated in favor of some combination of the remaining rules of *SD*. This was proved in Exercise 13.c in Section 5.4E. Hence *SD** is complete as well.

8. We used the fact that Conjunction Elimination is a rule of *SD* in proving (b) for 6.4.11, where we showed that if a sentence **P** & **Q** is a member of a set Γ^* that is maximally consistent in *SD*, then both **P** and **Q** are members of Γ^*.

9. First assume that some set Γ is truth-functionally consistent. Then obviously every finite subset of Γ is truth-functionally consistent as well, for all members of a finite subset of Γ are members of Γ, hence all are true on at least one truth-value assignment.

Now assume that some set Γ is truth-functionally inconsistent. If Γ is finite, then obviously at least one finite subset of Γ (namely, Γ itself) is truth-functionally inconsistent. If Γ is infinite, then, by Lemma 6.4.3, Γ is inconsistent in *SD*, and, by 6.4.6, some finite subset Γ' of Γ is inconsistent in *SD*—that is, for some sentence **P**, $\Gamma' \vdash \mathbf{P}$ and $\Gamma' \vdash {\sim}\, \mathbf{P}$. Hence, by Metatheorem 6.3.3, $\Gamma' \vDash \mathbf{P}$ and $\Gamma' \vDash {\sim}\, \mathbf{P}$, so Γ' is truth-functionally inconsistent; hence not every finite subset of Γ is truth-functionally consistent.

Section 7.2E

1.a. ‘The President’ is a singular term, ‘Democrat’ is not
x is a Democrat
(‘w’ or ‘y’ or ‘z’ may be used in place of ‘x’)

c. ‘Sarah’ and ‘Smith College’ are the singular terms
x attends Smith College
Sarah attends x
x attends y

e. The singular terms are ‘Charles’ and ‘Rita’
w and Rita are brother and sister
Charles and w are brother and sister
w and z are brother and sister

g. The singular terms are ‘2’, ‘4’, and ‘8’
x times 4 is 8
2 times x is 8
2 times 4 is y
x times y is 8
x times 4 is y
2 times x is y
x times y is z

i. The singular terms are ‘0’, ‘0’, and ‘0’
z plus 0 is 0
0 plus z is 0
0 plus 0 is z
w plus y is 0
w plus 0 is y
0 plus w is y
w plus y is z

2. Herman is larger than Herman.
Herman is larger than Juan.
Herman is larger than Antonio.
Juan is larger than Herman.
Juan is larger than Juan.
Juan is larger than Antonio.
Antonio is larger than Herman.
Antonio is larger than Juan.
Antonio is larger than Antonio.

Herman is to the right of Herman.
Herman is to the right of Juan.
Herman is to the right of Antonio.
Juan is to the right of Herman.
Juan is to the right of Juan.
Juan is to the right of Antonio.
Antonio is to the right of Herman.
Antonio is to the right of Juan.
Antonio is to the right of Antonio.

Herman is larger than Herman but smaller than Herman.
Herman is larger than Herman but smaller than Juan.
Herman is larger than Herman but smaller than Antonio.
Herman is larger than Juan but smaller than Herman.
Herman is larger than Juan but smaller than Juan.
Herman is larger than Juan but smaller than Antonio.
Herman is larger than Antonio but smaller than Herman.
Herman is larger than Antonio but smaller than Juan.
Herman is larger than Antonio but smaller than Antonio.

Juan is larger than Herman but smaller than Herman.
Juan is larger than Herman but smaller than Juan.
Juan is larger than Herman but smaller than Antonio.
Juan is larger than Juan but smaller than Herman.
Juan is larger than Juan but smaller than Juan.
Juan is larger than Juan but smaller than Antonio.
Juan is larger than Antonio but smaller than Herman.
Juan is larger than Antonio but smaller than Juan.
Juan is larger than Antonio but smaller than Antonio.

Antonio is larger than Herman but smaller than Herman.
Antonio is larger than Herman but smaller than Juan.
Antonio is larger than Herman but smaller than Antonio.
Antonio is larger than Juan but smaller than Herman.
Antonio is larger than Juan but smaller than Juan.
Antonio is larger than Juan but smaller than Antonio.
Antonio is larger than Antonio but smaller than Herman.
Antonio is larger than Antonio but smaller than Juan.
Antonio is larger than Antonio but smaller than Antonio.

EXERCISES 7.3E

1. The *PL* analogs of the sentences of English, in the same order given in the *Solution Manual* answers to exercise 7.2E 2, are

Lhh
Lhj
Lha
Ljh
Ljj
Lja
Lah
Laj
Laa

Rhh
Rhj
Rha
Rjh
Rjj
Rja
Rah
Raj
Raa

Shhh
Shhj
Shha
Shjh
Shjj
Shja
Shah
Shaj
Shaa

Sjhh
Sjhj
Sjha
Sjjh
Sjjj
Sjja
Sjah
Sjaj
Sjaa

Sahh
Sahj
Saha
Sajh
Sajj
Saja
Saah
Saaj
Saaa

2. a. Bai
c. Bbn
e. Beh
g. (Aph & Ahn) & Ank
i. Aih ≡ Aip
k. ([(Lap & Lbp) & (Lcp & Ldp)] & Lep) & ~ ([(Bap ∨ Bbp) ∨ (Bcp ∨ Bdp)] ∨ Bep)
m. (Tda & Tdb) & (Tdc & Tde)
o. ~ ([(Tab ∨ Tac) ∨ (Tad ∨ Tae)] ∨ Taa) & [(Lab & Lac) & (Lad & Lae)]

3. a. (Ia & Ba) & ~ Ra
c. (Bd & Rd) & Id
e. Ib ⊃ (Id & Ia)
g. Lab & Dac
i. ~ (Lca ∨ Dca) & (Lcd & Dcd)
k. Acb ≡ (Sbc & Rb)
m. (Sdc & Sca) ⊃ Sda
o. (Lcb & Lba) ⊃ (Dca & Sca)
q. Rd & ~ [Ra ∨ (Rb ∨ Rc)]

4. a. UD: Margaret, Todd, Charles, and Sarah
Gx: x is good at skateboarding
Lx: x likes skateboarding
Hx: x wears headgear
Kx: x wears knee pads
Rxy: x is more reckless than y (at skateboarding)
Sxy: x is more skillful than y (at skateboarding)
c: Charles
m: Margaret
s: Sarah
t: Todd

(Lm & Lt) & ~ (Gm ∨ Gt)
Gc & ~ Lc
Gs & Ls
[(Hm & Ht) & (Hc & Hs)] & [(Kc & Ks) & ~ (Km ∨ Kt)]
[(Rsm & Rst) & Rsc] & [(Scs & Scm) & Sct]

Note: it may be tempting to use a two-place predicate to symbolize being good at skateboarding, for example, ‘Gxy’, and another two-place predicate to symbolize liking skateboarding. So too we might use two-place predicates to symbolize wearing headgear and wearing kneepads. Doing so would require including skateboarding, headgear, and knee pads in the universe of discourse. But things are now a little murky. Skateboarding is more of an activity than a thing (although activities are often the “topics of conversation” as when we say that some people like, for example, hiking, skiing, and canoeing while others don’t). And while we might include all headgear and kneepads in our universe of discourse, we do not know which ones the characters in our passage wear, so we would be hard pressed to name the favored items.

Moreover, here there is no need to invoke these two-place predicates because here we are not asked to investigate logical relations that can only be expressed with two-place predicates. The case would be different if the passage included the sentence ‘If Sarah is good at anything she is good at sailing’ and we were asked to show that it follows from the passage that Sarah is good at sailing. (On the revised scenario we are told that Sarah is good at skateboarding, and that if she is good at anything—she is, skateboarding—she is good at sailing. So she is good at sailing. Here we are treating skateboarding as *something,* something Sarah is good at. But we will leave these complexities until we have fully developed the language *PL*.)

c. One appropriate symbolization key is

UD: Andrew, Christopher, Amanda
Hz: z is a hiker
Mz: z is a mountain climber
Kz: z is a kayaker
Sz: z is a swimmer
Lzw: z likes w
Nzw: z is nuts about w
a: Andrew
c: Christopher
m: Amanda

(Ha & Hc) & ~ (Ma ∨ Mc)
(Hm & Mm) & Km
(Ka ∨ Kc) & ~ (Ka & Kc)
~ [(Sa ∨ Sc) ∨ Sm]
((Lac & Lca) & [(Lam & Lma) & (Lmc & Lcm)]) & (Nma & Nam)

Section 7.4E

1.a. $(\forall z)Bz$
c. $\sim (\exists x)Bx$
e. $(\exists x)Bx \ \& \ (\exists x)Rx$
g. $(\exists z)Rz \supset (\exists z)Bz$
i. $(\forall y)By \equiv \sim (\exists y)Ry$

2.a. $(\exists x)Ox \ \& \ (\exists x)Ex$
c. $\sim (\exists x)Lxa$
e. $(\forall x)Gx$
g. $(\exists x)(Px \ \& \ Ex)$
i. $(\forall y)[(Py \ \& \ Lby) \supset Ey]$
k. $(\exists y)(Lby \ \& \ Lyc)$

3.a. $Pj \supset (\forall x)Px$
c. $(\exists y)Py \supset (Pj \ \& \ Pr)$
e. $\sim Pr \supset \sim (\exists x)Px$
g. $(Pj \supset Pr) \ \& \ (Pr \supset (\forall x)Px)$
i. $(\forall y)Sy \ \& \sim (\forall y)Py$
k. $(\forall x)Sx \supset (\exists y)Py$

Section 7.5E

1.a. A formula but not a sentence (an open sentence): the ‘z’ in ‘Zz’ is free.

c. A formula and a sentence.

e. A formula but not a sentence (an open sentence): the ‘x’ in ‘Fxz’ is free.

g. A formula and a sentence.

i. Not a formula. ‘∼ (∃x)’ is an expression of *SL*, but ‘(∼ ∃x)’ is not.

k. Not a formula. Since there is no ‘y’ in ‘Lxx’, ‘(∃y)Lxx’ is not a formula. Hence, neither is ‘(∃x)(∃y)Lxx’.

m. A formula and a sentence.

o. A formula but not a sentence (an open sentence): ‘w’ in ‘Fw’ is free.

2.a. A sentence. The subformulas are

$(\exists x)(\forall y)Byx$	$(\exists x)$
$(\forall y)Byx$	$(\forall y)$
Byx	None

c. Not a sentence. The ‘x’ in ‘(Bg ⊃ Fx)’ is free. The subformulas are

(∀x)(~ Fx & Gx) ≡ (Bg ⊃ Fx)	≡
(∀x)(~ Fx & Gx)	(∀x)
Bg ⊃ Fx	⊃
~ Fx & Gx	&
~ Fx	~
Gx	None
Bg	None
Fx	None

e. Sentence. The subformulas are

~ (∃x)Px & Rab	&
~ (∃x)Px	~
Rab	None
(∃x)Px	(∃x)
Px	None

g. Sentence. The subformulas are

~ [~ (∀x)Fx ≡ (∃w) ~ Gw] ⊃ Maa	⊃
~ [~ (∀x)Fx ≡ (∃w) ~ Gw]	~
Maa	None
~ (∀x)Fx ≡ (∃w) ~ Gw	≡
~ (∀x)Fx	~
(∃w) ~ Gw	(∃w)
(∀x)Fx	(∀x)
Fx	None
~ Gw	~
Gw	None

i. Sentence. The subformulas are

~ ~ ~ (∃x)(∀z)(Gxaz ∨ ~ Hazb)	~
~ ~ (∃x)(∀z)(Gxaz ∨ ~ Hazb)	~
~ (∃x)(∀z)(Gxaz ∨ ~ Hazb)	~
(∃x)(∀z)(Gxaz ∨ ~ Hazb)	(∃x)
(∀z)(Gxaz ∨ ~ Hazb)	(∀z)
Gxaz ∨ ~ Hazb	∨
Gxaz	None
~ Hazb	~
Hazb	None

k. Sentence. The subformulas are

(∃x)[Fx ⊃ (∀w)(~ Gx ⊃ ~ Hwx)]	(∃x)
Fx ⊃ (∀w)(~ Gx ⊃ ~ Hwx)	⊃
Fx	None
(∀w)(~ Gx ⊃ ~ Hwx)	(∀w)
~ Gx ⊃ ~ Hwx	⊃
~ Gx	~
~ Hwx	~
Gx	None
Hwx	None

m. A sentence. The subformulas are

(Hb ∨ Fa) ≡ (∃z)(~ Fz & Gza)	≡
Hb ∨ Fa	∨
(∃z)(~ Fz & Gza)	(∃z)
Hb	None
Fa	None
~ Fz & Gza	&
~ Fz	~
Gza	None
Fz	None

3.a.	(∀x)(Fx ⊃ Ga)	Quantified
c.	~ (∀x)(Fx ⊃ Ga)	Truth-functional
e.	~ (∃x)Hx	Truth-functional
g.	(∀x)(Fx ≡ (∃w)Gw)	Quantified
i.	(∃w)(Pw ⊃ (∀y)(Hy ≡ ~ Kyw))	Quantified
k.	~ [(∃w)(Jw ∨ Nw) ∨ (∃w)(Mw ∨ Lw)]	Truth-functional
m.	(∀z)Gza ⊃ (∃z)Fz	Truth-functional
o.	(∃z) ~ Hza	Quantified
q.	(∀x) ~ Fx ≡ (∀z) ~ Hza	Truth-functional

4.a. Maa & Fa
c. ~ (Ca ≡ ~ Ca)
e. (Fa & ~ Gb) ⊃ (Bab ∨ Bba)
g. ~ (∃z)Naz ≡ (∀w)(Mww & Naw)
i. Fab ≡ Gba
k. ~ (∃y)(Hay & Hya)
m. (∀y)[(Hay & Hya) ⊃ (∃z)Gza]

5.a.	$(\forall y)Ray \supset Byy$	No
c.	$(\forall y)(Rwy \supset Byy)$	No
e.	$(\forall y)(Ryy \supset Byy)$	No
g.	$(Ray \supset Byy)$	No
i.	$Rab \supset Bbb$	No

6.a.	$(\forall y) \sim Ray \equiv Paa$	Yes
c.	$(\forall y) \sim Ray \equiv Pba$	No
e.	$(\forall y)(\sim Ryy \equiv Paa)$	No
g.	$(\forall y) \sim Raw \equiv Paa$	No

Section 7.6E

1.a.	A-sentence	$(\forall y)(Py \supset Cy)$
c.	O-sentence	$(\exists w)(Dw \ \& \sim Sw)$
e.	I-sentence	$(\exists z)(Nz \ \& \ Bz)$
g.	E-sentence	$(\forall x)(Px \supset \sim Sx)$
i.	A-sentence	$(\forall w)(Pw \supset Mw)$
k.	A-sentence	$(\forall y)(Sy \supset Cy)$
m.	E-sentence	$(\forall y)(Ky \supset \sim Sy)$
o.	E-sentence	$(\forall y)(Qy \supset \sim Zy)$

2.a. $(\forall y)(By \supset Ly)$
c. $(\forall z)(Rz \supset \sim Lz)$
e. $(\exists x)Bx \ \& \ (\exists x)Rx$
g. $[(\exists z)Bz \ \& \ (\exists z)Rz] \ \& \sim (\exists z)(Bz \ \& \ Rz)$
i. $(\exists y)By \ \& \ [(\exists y)Sy \ \& \ (\exists y)Ly]$
k. $(\forall w)(Cw \supset Rw) \ \& \sim (\forall w)(Rw \supset Cw)$
m. $(\forall y)Ry \vee [(\forall y)By \vee (\forall y)Gy]$
o. $(\exists w)(Rw \ \& \ Sw) \ \& \ (\exists w)(Rw \ \& \sim Sw)$
q. $(\exists x)Ox \ \& \ (\forall y)(Ly \supset \sim Oy)$

3.a. An I-sentence and the corresponding O-sentence of *PL* can both be true. Consider the English sentences ‘Some positive integers are even’ and ‘Some positive integers are not even’. Where the UD is positive integers and ‘Ex’ is interpreted as ‘x is even’, these can be symbolized as ‘$(\exists x)Ex$’ and ‘$(\exists x) \sim Ex$’, respectively, and both sentences of *PL* are true.

An I-sentence and an O-sentence can also both be false. Consider ‘Some tiggers are fast’ and ‘Some tiggers are not fast’. Where the UD is mammals, ‘Tx’ is interpreted as ‘x is a tigger’ and ‘Fx’ as ‘x is fast’, these become, respectively, ‘$(\exists x)(Tx \ \& \ Fx)$’ and ‘$(\exists x)(Tx \ \& \sim Fx)$’ As there are no tiggers, both sentences of *PL* are false. Note, however, that there cannot be an I-sentence and a corresponding O-sentence of the sorts $(\exists \mathbf{x})\mathbf{A}$ and $(\exists \mathbf{x}) \sim \mathbf{A}$, where $\mathbf{A}$ is anj atomic formula and both the I-sentence and the O-sentence are false. For however $\mathbf{A}$ is interpreted, either there is something that satisfies it, or there is not. In the first instance $(\exists \mathbf{x})\mathbf{A}$ is true, in the second $(\exists \mathbf{x}) \sim \mathbf{A}$ is true.

Section 7.7E

1.a. $(\forall z)(Pz \supset Hz)$
c. $(\exists z)(Pz \;\&\; Hz)$
e. $(\forall w)[(Hw \;\&\; Pw) \supset \sim Iw]$
g. $\sim (\forall x)[(Px \vee Ix) \supset Hx]$
i. $(\forall y)[(Iy \;\&\; Hy) \supset Ry]$
k. $(\exists z)Iz \supset Ih$
m. $(\exists w)Iw \supset (\forall x)(Rx \supset Ix)$
o. $\sim (\exists y)[Hy \;\&\; (Py \;\&\; Iy)]$
q. $(\forall z)(Pz \supset Iz) \supset \sim (\exists z)(Pz \;\&\; Hz)$
s. $(\forall w)(Rw \supset [(Lw \;\&\; Iw) \;\&\; \sim Hw])$

2.a. $(\forall w)(Lw \supset Aw)$
c. $(\forall x)(Lx \supset Fx) \;\&\; (\forall x)(Tx \supset \sim Fx)$
e. $(\exists y)[(Fy \;\&\; Ly) \;\&\; Cdy]$
g. $(\forall z)[(Lz \vee Tz) \supset Fz]$
i. $(\exists w)(Tw \;\&\; Fw) \;\&\; \sim (\forall w)(Tw \supset Fw)$
k. $(\forall x)[(Lx \;\&\; Cbx) \supset (Ax \;\&\; \sim Fx)]$
m. $(\exists z)(Lz \;\&\; Fz) \supset (\forall w)(Tw \supset Fw)$
o. $\sim Fb \;\&\; Bb$

3.a. $(\forall x)(Ex \supset Yx)$
c. $(\exists y)(Ey \;\&\; Yy) \;\&\; \sim (\forall y)(Ey \supset Yy)$
e. $(\exists z)(Ez \;\&\; Yz) \supset (\forall x)(Lx \supset Yx)$
g. $(\forall w)[(Ew \;\&\; Sw) \supset Yw]$
i. $(\forall w)[(Lw \;\&\; Ew) \supset (Yw \;\&\; Iw)]$
k. $(\forall x)[(Ex \vee Lx) \supset (Yx \supset Ix)]$
m. $\sim (\exists z)[(Pz \;\&\; \sim Iz) \;\&\; Yz]$
o. $(\forall x)[(Ex \;\&\; Rxx) \supset Yx]$
q. $(\forall x)([Ex \vee Lx) \;\&\; (Rx \vee Yx)] \supset Rxx)$
s. $(\forall z)([Yz \;\&\; (Lz \;\&\; Ez)] \supset Rzz)$

4.a. $(\forall x)[Px \supset (Ux \;\&\; Ox)]$
c. $(\forall z)[Az \supset \sim (Oz \vee Uz)]$
e. $(\forall w)(Ow \equiv Uw)$
g. $(\exists y)(Py \;\&\; Uy) \;\&\; (\forall y)[(Py \;\&\; Ay) \supset \sim Uy]$
i. $(\exists z)[Pz \;\&\; (Oz \;\&\; Uz)] \;\&\; (\forall x)[Sx \supset (Ox \;\&\; Ux)]$
k. $((\exists x)(Sx \;\&\; Ux) \;\&\; (\exists x)(Px \;\&\; Ux)) \;\&\; \sim (\exists x)(Ax \;\&\; Ux)$

5.a. Two is prime and three is prime.
c. There is an integer that is even and there is an integer that is odd.
e. Each integer is either even or odd.
g. There is an integer that is not larger than one. [Note: that integer is one itself.]
i. Each integer is such that if it is even then it is evenly divisible by two.
k. Every integer is evenly divisible by one.

m. An integer is evenly divisible by two if and only if it is even.

o. If one is larger than some integer then it is larger than every integer.

q. No integer is prime and evenly divisible by four.

Section 7.8E

1.a. (∃y)[Sy & (Cy & Ly)]

c. ~ (∀w)[(Sw & Lw) ⊃ Cw]

e. ~ (∀x)[(∃y)(Sy & Sxy) ⊃ Sx]

g. ~ (∀x)[(∃y)(Sy & (Dxy ∨ Sxy)) ⊃ Sx]

i. (∀z)[(Sz & (∃w)(Swz ∨ Dwz)) ⊃ Lz]

k. Sr ∨ (∃y)(Sy & Dry)

m. (Sr & (∀z)[(Dzr ∨ Szr) ⊃ Sz]) ∨ (Sj & (∀z)[(Dzj ∨ Szj) ⊃ Sz])

2.a. (∀x)[Ax ⊃ (∃y)(Fy & Exy)] & (∀x)[Fx ⊃ (∃y)(Ay & Exy)]

c. ~(∃y)(Fy & Eyp)

e. ~(∃y)(Fy & Eyp) & (∃y)(Cy & Eyp)

g. ~ (∃w)(Aw & Uw) & (∃w)(Aw & Fw)

i. (∃w)[(Aw & ~ Fw) & (∀y)[(Fy & Ay) ⊃ Ewy]]

k. (∃z)[Fz & (∀y)(Ay ⊃ Dzy)] & (∃z)[Az & (∀y)(Fy ⊃ Dzy)]

m. (∀x)[(∀y)Dxy ⊃ (Px ∨ (Ax ∨ Ox))]

3.a. (∀x)[Px ⊃ (∃y)(Syx & Bxy)]

c. (∀y)[(Py & (∀z)Bzy) ⊃ (∀w)(Swy ⊃ Byw)]

e. (∀w)(∀x)[(Pw & Sxw) ⊃ Bwx] ⊃ (∀z)(Pz ⊃ Wz)

g. (∀x)(∀y)([(Px & Syx) & Bxy] ⊃ (~ Nxy & ~ Lyx))

i. (∃y)[Py & (∀z)(Pz ⊃ Byz)]

k. (∀z)((Pz & Uz) ⊃ [(∀w)(Swz ⊃ Bzw) ∨ (∀w)(Swz ⊃ Gzw)])

m. (∀w)(∀x)([(Pw & Sxw) & (Bwx & Bxw)] ⊃ (Ww & Wx))

o. (∃x)(∃y)[(Px & Syx) & ~ Axty]

q. (∀y)(∀z)([(Py & Szy) & ~ Lzy] ⊃ (~ Nzy & Bzy))

4.a. Hildegard sometimes loves Manfred.

c. Manfred sometimes loves Hildegard and Manfred always loves Siegfried.

e. If Manfred ever loves himself, then he does so whenever Hildegard loves him.

g. There is someone no one ever loves.

i. There is a time at which someone loves everyone.

k. There is always someone who loves everyone.

m. No one loves anyone all the time.

o. Everyone loves, at some time, himself or herself.

5.a. An even integer times any integer is even.

c. If the sum of a pair of integers is even, then either both integers are even or both are odd.

e. There is no prime that is larger than every prime.

g. There are no primes such that their product is prime.

i. There is a prime such that it times any prime is even.

k. The product of a pair of integers is odd if and only if both members of the pair are odd.

m. If a pair of integers are both odd, then their product is odd and their sum is even.

o. The sum of an odd integer and an even integer is odd, and their product is even.

q. There is an integer that is larger than one, that three is larger than, and that is prime and even.

Section 7.9E

1.a. $(\forall x)[(Wx \,\&\, \sim x = d) \supset Sx]$

c. $(\forall x)[(Wx \,\&\, \sim x = d) \supset [Sx \vee (\exists y)[Sy \,\&\, (Dxy \vee Sxy)]]]$

e. $[Sdj \,\&\, (\forall x)(Sxj \supset x = d)] \,\&\, \sim (\exists x)Dxj$

g. $(\exists x)[(Sxr \,\&\, Sxj) \,\&\, (\forall y)[(Syr \vee Syj) \supset y = x]]$

i. $(\exists x)(\exists y)[((Dxr \,\&\, Dyr) \,\&\, (Sx \,\&\, Sy)) \,\&\, \sim x = y]$

k. $(\exists x)[(Sxj \,\&\, Sx) \,\&\, (\forall y)(Syj \supset y = x)] \,\&\, (\exists x)(\exists y)(([(Sx \,\&\, Sy) \,\&\, (Dxj \,\&\, Dyj)] \,\&\, \sim x = y) \,\&\, (\forall z)[Dzj \supset (z = x \vee z = y)])$

2.a. Every positive integer is less than some positive integer [or] There is no largest positive integer.

c. There is positive integer than which no integer is less.

e. 2 is even and prime, and it is the only positive integer that is both even and prime.

g. The product of any pair of odd positive integers is itself odd.

i. If either of a pair of positive integers is even, their product is even.

k. There is exactly one prime that is greater than 5 and less than 9.

3.a.	$(\forall x)(\forall y)(Nxy \supset Nyx)$	Symmetric only
c.		Neither reflexive, nor symmetric, nor transitive
e.	$(\forall x)(\forall y)(Rxy \supset Ryx)$ $(\forall x)(\forall y)(\forall z)[(Rxy \,\&\, Ryz) \supset Rxz]$	Symmetric and transitive
g.	$(\forall x)Txx$ $(\forall x)(\forall y)(\forall z)[(Txy \,\&\, Tyz) \supset Txz]$	Transitive and reflexive (in UD: Physical objects)
i.	$(\forall x)(\forall y)(Exy \supset Eyx)$ $(\forall x)Exx$	Symmetric and reflexive (in UD: People)
k.	$(\forall x)Wxx$ $(\forall x)(\forall y)(Wxy \supset Wyx)$ $(\forall x)(\forall y)(\forall z)[(Wxy) \,\&\, Wyz) \supset Wxz]$	Symmetric, transitive, and reflexive (in UD: Physical objects)
m.	$(\forall x)(\forall y)(\forall z)[(Axy \,\&\, Ayz) \supset Axz]$	Transitive only
o.	$(\forall x)Lxx$ $(\forall x)(\forall y)(Lxy \supset Lyx)$ $(\forall x)(\forall y)(\forall z)[(Lxy \,\&\, Lyz) \supset Lxz]$	Symmetric, transitive, and reflexive (in UD: People)

4.a. Sjc
c. $Sjc \mathbin{\&} (\forall x)[(Sxc \mathbin{\&} {\sim}\, x = j) \supset Ojx]$
e. $(\exists x)[(Dxd \mathbin{\&} (\forall y)[(Dyd \mathbin{\&} {\sim}\, y = x) \supset Oxy]) \mathbin{\&} Px]$
g. $Dcd \mathbin{\&} (\forall x)[(Dxd \mathbin{\&} {\sim}\, x = c) \supset Ocx]$
i. $(\exists x)[(Sxh \mathbin{\&} (\forall y)[(Syh \mathbin{\&} {\sim}\, y = x) \supset Txy]) \mathbin{\&} Mcx]$
k. $(\exists x)[(Bx \mathbin{\&} (\forall y)(By \supset y = x)) \mathbin{\&}$
$(\exists w)((Mx \mathbin{\&} (\forall z)(Mz \supset z = w)) \mathbin{\&} x = w)]$
m. $(\exists x)[(Mxc \mathbin{\&} Bxj) \mathbin{\&} (\forall w)(Bwj \supset x = w)]$

5.a. ${\sim}\, (\exists y)a = f(y)$
c. $(\exists x)(Px \mathbin{\&} Ex)$
e. $(\forall x)(\exists y)y = f(x)$
g. $(\forall y)(Oy \supset Ef(y))$
i. $(\forall x)(\forall y)[Ot(x,y) \supset Et(f(x), f(y))]$
k. $(\forall x)(\forall y)[Os(x,y) \supset [(Ox \mathbin{\&} Ey) \vee (Oy \mathbin{\&} Ex)]]$
m. $(\forall x)(\forall y)[(Px \mathbin{\&} Py) \supset {\sim}\, Pt(x,y)]$
o. $(\forall z)[(Ez \supset Eq(z)) \mathbin{\&} (Oz \supset Oq(z)]$
q. $(\forall x)[Ox \supset Ef(q(x))]$
s. $(\forall x)[(Px \mathbin{\&} {\sim}\, x = b) \supset Os(b,x)]$
u. $(\exists x)(\exists y)[(Px \mathbin{\&} Py) \mathbin{\&} t(x,y) = f(s(x,y))]$

CHAPTER EIGHT

Section 8.1E

1.a. **F**
c. **T**
e. **F**
g. **T**

2.a. **T**
c. **T**
e. **F**
g. **F**

3.a. One interpretation is

UD: Set of people
Nxy: x is the mother of y
a: Jane Doe
d: Jay Doe

c. One interpretation is

UD: Set of U.S. cities
Lx: x is in California
Cxy: x is to the north of y
h: San Francisco
m: Los Angeles

e. One interpretation is

UD: Set of positive integers
Mx: x is odd
Nx: x is even
a: 1
b: 2

4.a. One interpretation is

UD: Set of positive integers
Cxy: x equals y squared
r: 2
s: 3

c. One interpretation is

UD: Set of people
Lx: x is a lion
i: Igor Stravinsky
j: Jesse Winchester
m: Margaret Mead

e. One interpretation is

UD: Set of positive integers
Jx: x is even
a: 1
b: 2
c: 3
d: 4

5.a. One interpretation is

UD: Set of people
Fxy: x is the mother of y
a: Liza Minelli
b: Judy Garland (Liza Minelli’s mother)

On this interpretation, ‘Fab ⊃ Fba’ is true, and ‘Fba ⊃ Fab’ is false.

c. One interpretation is

UD: Set of planets
Cxyz: the orbit of x is between the orbit of y and the orbit of z
Mx: x is inhabited by human life
a: Earth
p: Venus
q: Pluto
r: Mars

On this interpretation, ‘~ Ma ∨ Cpqr’ is false, and ‘Capq ∨ ~ Mr’ is true.

e. One interpretation is

UD: Set of positive integers
Lxy: x is less than y
Mxy: x equals y
j: 1
k: 1

On this interpretation the first sentence is true and the second false.

6.a. Suppose that ‘Ba’ is true on some interpretation. Then ‘$Ba \vee \sim Ba$’ is true on that interpretation. Suppose that ‘Ba’ is false on some interpretation. Then ‘$\sim Ba$’ is true on that interpretation, and so is ‘$Ba \vee \sim Ba$’. Since on any interpretation ‘Ba’ is either true or false, we have shown that ‘$Ba \vee \sim Ba$’ is true on every interpretation.

7.a. False. For consider any person w who is over 40 years old. It is true that that person is over 40 years old but false that some person is her own sister. So that person w is *not* such that if w is over 40 years old then some person is her own sister.

c. False. The sentence says that there is at least one person x such that every person y is either a child or a brother of x, which is obviously false.

e. True. The antecedent, ‘$(\exists x)Cx$’, is true. At least one person is over 40 years old. And the consequent, ‘$((\exists x)(\exists y)Fxy \supset (\exists y)By)$’, is also true: ‘$(\exists x)(\exists y)Fxy$’ is true, and ‘$(\exists y)By$’ is true.

g. True. The antecedent, ‘$(\forall x)Bx$’, is false, so the conditional sentence is true.

i. True. The sentence says that there is at least one person x such that either x is over 40 years old or x and some person y are sisters and y is over 40 years old. Both conditions are true.

8.a. True. Every U.S. president held office after George Washington’s first term. Note that for the sentence to be true, George Washington too must have held office after George Washington’s first term of office. He did—he was in office for two terms.

c. True. George Washington was the first U.S. president, and at least one U.S. president y held office after Washington.

e. True. Each U.S. president y is such that if y is a U.S. citizen (which every U.S. president y is) then at least one U.S. president held office before or after y’s first term.

g. False. Every U.S. president x held office after George Washington’s first term, but, for any such president x, no non-U.S. citizen has held office before x (because every U.S. president *is* a U.S. citizen).

i. True (in 2003!). The sentence says that a disjunction is not the case and therefore that each disjunct is false. The first disjunct, ‘Bg’, is false—George Washington was not a female. The second disjunct, which says that there is a U.S. president who held office after every U.S. president’s first term of office, is false (there is no one yet who has held office after George W. Bush’s first term).

9.a. True. The first conjunct, ‘Bb’, is true. The second conjunct is also true since no positive integer that is greater than 2 is equal to 2.

c. True. No positive integer x is equal to any number than which it is greater.

e. True. The antecedent is true since it is not the case that every positive integer is greater than every positive integer. But ‘Mcba’ is also true: $3 - 2 = 1$.

g. True. No positive integer z that is even is such that the result of subtracting 1 from z is also even.

i. False. Not every positive integer (in fact, *no* positive integer) is such that it equals itself if and only if there are not two positive integers of which it is the difference. Every positive integer equals itself, but every positive integer is also the difference between two positive integers.

Section 8.2E

1.a. The sentence is false on the following interpretation:

UD: Set of positive integers
Fx: x is divisible by 4
Gx: x is even

Every positive integer that is divisible by 4 is even, but not every positive integer is even.

c. The sentence is false on the following interpretation:

UD: Set of positive integers
Bxy: x is less than y

Every positive integer is less than at least one positive integer, but there is no single positive integer that every positive integer is less than.

e. The sentence is false on the following interpretation:

UD: Set of positive integers
Fx: x is odd
Gx: x is prime

The antecedent, ‘$(\forall x)Fx \supset (\forall w)Gw$’, is true since *its* antecedent, ‘$(\forall x)Fx$’, is false. But the consequent, ‘$(\forall z)(Fz \supset Gz)$’, is false since at least one odd positive integer is not prime (the integer 9, for example).

g. The sentence is false on the following interpretation:

UD: Set of positive integers
Gx: x is negative
Fxy: x equals y

No positive integer is negative, but not every positive integer is such that if it equals itself (which every one does) then it is negative.

2.a. The sentence is true on the following interpretation:

UD: Set of positive integers
Bxy: x equals y

The sentence to the left of '≡' is true since it is not the case that all positive integers equal one another; and the sentence to the right of '≡' is true since each positive integer is equal to itself.

c. The sentence is true on the following interpretation:

UD: Set of positive integers
Fx: x is odd
Gx: x is even

At least one positive integer is odd, and at least one positive integer is even, but no positive integer is both odd and even.

e. The sentence is true on the following interpretation:

UD: Set of positive integers
Fx: x is negative
Gx: x is odd

Trivially, every negative positive integer is odd since no positive integer is negative; and every positive integer that is odd is not negative.

g. The sentence is true on the following interpretation:

UD: Set of positive integers
Bx: x is prime
Hx: x is odd

The antecedent is false—not every positive integer is such that it is prime if and only if it is odd, and the consequent is true—at least one positive integer is both prime and odd.

i. The sentence is true on the following interpretation:

UD: Set of positive integers
Bxy: x is less than y

The less-than relation is transitive, making the first conjunct true; for every positive integer there is a greater one, making the second conjunct true; and the less-than relation is irreflexive, making the third conjunct true.

3.a. The sentence is true on the following interpretation:

UD: Set of positive integers
Fx: x is odd
Gx: x is prime

At least one positive integer is both odd and prime, but also at least one positive integer is neither odd nor prime.

The sentence is false on the following interpretation:

UD: Set of positive integers
Fx: x is positive
Gx: x is prime

At least one positive integer is both positive and prime, but no positive integer is neither positive nor prime.

c. The sentence is true on the following interpretation:

UD: Set of positive integers
Bxy: x is evenly divisible by y
n: the number 9

The antecedent, '$(\forall x)$Bnx', is false on this interpretation; 9 is not evenly divisible by every positive integer.

The sentence is false on the following interpretation:

UD: Set of positive integers
Bxy: x is less than or equal to y
n: the number 1

The number 1 is less than or equal to every positive integer, so the antecedent is true and the consequent false.

e. The sentence is true on the following interpretation:

UD: Set of positive integers
Nxy: x equals y

Each positive integer x is such that each positive integer w that is equal to x is equal to itself.

The sentence is false on the following interpretation:

UD: Set of positive integers
Nxy: x is greater than y

No positive integer x is such that every positive integer w that is greater or smaller than x is greater than itself.

g. The sentence is true on the following interpretation:

UD: Set of positive integers
Cx: x is greater than 0
Dx: x is prime

Every positive integer is either greater than 0 or prime (because every positive integer is greater than 0), and at least one positive integer is both greater than 0 and prime. The biconditional is therefore true on this interpretation.

The sentence is false on the following interpretation:

UD: Set of positive integers
Cx: x is even
Dx: x is odd

Every positive integer is either even or odd, but no positive integer is both. The biconditional is therefore false on this interpretation.

4.a. If the antecedent is true on an interpretation, then at least one member x of the UD, let's assume a, stands in the relation B to every member y of the UD. But then it follows that for every member y of the UD, there is at least one member x that stands in the relation B to y—namely, a. So the consequent is also true. If the antecedent is false on an interpretation, then the conditional is trivially true. So the sentence is true on every interpretation.

c. If 'Fa' is true on an interpretation, then 'Fa ∨ [(∀x)Fx ⊃ Ga]' is true. If 'Fa' is false on an interpretation, then '(∀x)Fx' is false, making '(∀x)Fx ⊃ Ga' true. Either way, the disjunction is true.

e. If '(∃x)Hx' is true on an interpretation, then the disjunction is true on that interpretation. If '(∃x)Hx' is false on an interpretation, then no member of the UD is H. In this case, every member of the UD is such that if it is H (which it is not) then it is J, and so the second disjunct is true, making the disjunction true as well. Either way, then, the disjunction is true.

5.a. No member of any UD is such that it is in the extension of 'B' if and only if it isn't in the extension of 'B'. So the existentially quantified sentence is false on every interpretation.

c. The second conjunct is true on an interpretation if and only if no member of the UD is G and no member of the UD is not F—that is, every member of the UD *is* F. But then the first conjunct must be false, because its antecedent is true but its consequent is false. Thus there is no interpretation on which the entire conjunction is true; it is quantificationally false.

e. The third conjunct is true on an interpretation if and only if at least one member **u** of the UD is A but is not C. For the first conjunct to be true, **u** must also be B since it is A; and for the second conjunct to be true, **u** must also be C since it is B. But that means that the conjunction is true if and only if at least one member **u** of the UD is both C and not C. This latter is impossible; so there is no interpretation on which the sentence is true, i.e., it is quantificationally false.

6.a. The sentence is quantificationally indeterminate. It is true on the interpretation

UD: Set of positive integers
Gx: x is odd
Hx: x is even

since at least one positive integer is odd and at least one is even, and at least one positive integer (in fact, every positive integer) is not both odd and even.

The sentence is false on the interpretation

UD: Set of positive integers
Gx: x is less than zero
Hx: x is even

since the first conjunct is false: no positive integer is less than zero.

c. The sentence is quantificationally true. If every member of the UD that is F is also G, then every member of the UD that fails to be G must also fail to be F.

e. The sentence is quantificationally indeterminate. It is true on the interpretation

UD: Set of positive integers
Dx: x is odd
Hxy: x is greater than or equal to y

because the consequent, which says that there is a positive integer z such that every odd positive integer is greater than or equal to z, is true. The positive integer 1 satisfies this condition.

The sentence is false on the interpretation

UD: Set of positive integers
Dx: x is odd
Hxy: x equals y

because the antecedent, which says that for every odd positive integer there is at least one positive integer to which it is equal, is true; but the antecedent, which says that there is some one positive integer to which every odd positive integer is equal, is false.

Section 8.3E

1.a. The first sentence is false and the second true on the following interpretation:

UD: Set of positive integers
Fx: x is odd
Gx: x is prime
a: the number 4

Some positive integer is odd and the number 4 is not prime, so ‘$(\exists x)Fx \supset Ga$’ is false. But any even positive integer is such that if that integer is odd (which it is not) then the number 4 is prime; so ‘$(\exists x)(Fx \supset Ga)$’ is true.

c. The first sentence is false and the second true on the following interpretation:

UD: Set of integers
Fx: x is a multiple of 2
Gx: x is an odd number

It is false that either every integer is a multiple of 2 or every integer is odd, but it is true that every integer is either a multiple of 2 or odd.

e. The first sentence is false and the second true on the following interpretation:

UD: Set of positive integers
Fx: x is odd
Gx: x is prime

An odd prime (e.g., the number 3) is not such that it is even if and only if it is prime. But ‘$(\exists x)Fx \equiv (\exists x)Gx$’ is true since ‘$(\exists x)Fx$’ and ‘$(\exists x)Gx$’ are both true.

g. The first sentence is true and the second false on the following interpretation:

UD: Set of positive integers
Bx: x is less than 5
Dxy: x is divisible by y without remainder

The number 1 is less than 5 and divides every positive integer without remainder. But ‘$(\forall x)(Bx \supset (\forall y)Dyx)$’ is false, for 2 is less than 5 but does not divide any odd number without remainder.

i. The first sentence is false and the second true on the following interpretation:

UD: set of positive integers
Fx: x is odd
Kxy: x is smaller than y

The number 1 does not satisfy the condition that if it is odd (which it is) then there is a positive integer that is smaller than it. But at least one positive integer does satisfy the condition—in fact, all other positive integers do.

2.a. Suppose that ‘$(\forall x)Fx \supset Ga$’ is true on an interpretation. Then either ‘$(\forall x)Fx$’ is false or ‘Ga’ is true. If ‘$(\forall x)Fx$’ is false, then some member of the UD is not in the extension of ‘F’. But then that object is trivially such that if it is F (which it is not) then a is G. So ‘$(\exists x)(Fx \supset Ga)$’ is true. If ‘$Ga$’ is true, then trivially every member x of the UD is such that if x is F then a is G; so ‘$(\exists x)(Fx \supset Ga)$’ is true in this case as well.

Now suppose that ‘$(\forall x)Fx \supset Ga$’ is false on some interpretation. Then ‘$(\forall x)Fx$’ is true, and ‘Ga’ is false. Every object in the UD is then in the extension of ‘F’; hence no member x is such that if it is F (which it is) then a is G (which is false). So ‘$(\exists x)(Fx \supset Ga)$’ is false as well.

c. Suppose that ‘$(\exists x)(Fx \vee Gx)$’ is true on an interpretation. Then at least one member of the UD is either in the extension of ‘F’ or in the extension of ‘G’. This individual therefore does not satisfy ‘$\sim Fy \& \sim Gy$’, and so ‘$(\forall y)(\sim Fy \& \sim Gy)$’ is false and its negation true.

Now suppose that ‘$(\exists x)(Fx \vee Gx)$’ is false on an interpretation. Then no member of the UD satisfies ‘$Fx \vee Gx$’—no member of the UD is in the extension of ‘F’ or in the extension of ‘G’. In this case, every member of the UD satisfies ‘$\sim Fy \& \sim Gy$’; so ‘$(\forall y)(\sim Fy \& \sim Gy)$’ is true and its negation false.

e. Suppose that ‘$(\forall x)(\forall y)Gxy$’ is true on an interpretation. Then each pair of objects in the UD is in the extension of ‘G’. But then ‘$(\forall y)(\forall x)Gxy$’ must also be true. The same reasoning establishes the reverse.

3.a. The sentences are not quantificationally equivalent. The first sentence is true and the second false on the following interpretation:

UD: Set of positive integers
Fx: x is greater than 4
Gx: x is less than 10

At least one positive integer is either greater than 4 or less than 10, but it is false that every positive integer fails to be both greater than 4 and less than 10.

c. The sentences are not quantificationally equivalent. The first sentence is false and the second true on the following interpretation:

UD: Set of positive integers
Gxy: x equals y

It is false that each pair of positive integers is such that either the first equals the second or vice versa, but it is true that each pair of positive integers is such that either the first member equals itself (which is always true) or it is equal to the second.

4.a. All the set members are true on the following interpretation:

UD: Set of positive integers
Bx: x is odd
Cx: x is prime

At least one positive integer is odd, and at least one positive integer is prime, and some positive integers are neither odd nor prime.

c. All the set members are true on the following interpretation:

UD: Set of positive integers
Fx: x is greater than 10
Gx: x is greater than 5
Nx: x is smaller than 3
Mx: x is smaller than 5

Every positive integer that is greater than 10 is greater than 5, every positive integer that is smaller than 3 is smaller than 5, and no positive integer that is greater than 5 is also smaller than 5.

e. All the set members are true on the following interpretation:

UD: Set of positive integers
Nx: x is negative
Mx: x equals 0
Cxy: x is greater than 7

The two sentences are trivially true, the first because no positive integer is negative and the second because no positive integer equals 0.

g. All the set members are true on the following interpretation:

UD: Set of positive integers
Nx: x is prime
Mx: x is an even number

The first sentence is true because 3 is prime but not even. Hence not all primes are even numbers. The second is true because any nonprime integer is such that if it is prime (which it is not) then it is even. Hence it is false that all positive integers fail to satisfy this condition.

i. All the set members are true on the following interpretation:

UD: Set of positive integers
Fxy: x evenly divides y
Gxy: x is greater than y
a: 1

At least one positive integer is evenly divisible by 1, at least one positive integer is such that 1 is not greater than that integer, and every positive integer is either evenly divisible by 1 or such that 1 is greater than it.

5.a. If the set is quantificationally consistent, then there is an interpretation on which both set members are true. But if '$(\exists x)(Bx \& Cx)$' is true on an interpretation, then at least one member x of the UD is in the extensions of both 'B' and 'C'. That member is *not* neither B nor C, so, if '$(\exists x)(Bx \& Cx)$' is true, then '$(\forall x) \sim (Bx \vee Cx)$' is false. There is no interpretation on which both set members are true.

c. If the first set member is true on an interpretation, then every pair x and y of members of the UD is such that either x stands in the relation B to y or y stands in the relation B to x. In particular, each pair consisting of a member of the UD and itself must satisfy the condition and so must stand in the relation B to itself. This being so, the second set member is false on such an interpretation. Thus there can be no interpretation on which both set members are true.

e. If the first sentence is true on an interpretation, then there is at least one member of the UD that stands in the relation G to every member of the UD. In that case it is false that every pair of members of the UD fail to satisfy 'Gxy', so the second sentence must be false. Thus there can be no interpretation on which both set members are true.

6.a. The set is quantificationally inconsistent. If the third member is true, then something in the UD is F. If the first member is also true, then, because the antedent will be true, the consequent will also be true: everything in the UD will be F. But then the second sentence must be false: there is nothing that is not F. Thus there can be no interpretation on which all three set members are true.

c. The set is quantificationally consistent, as the following interpretation shows:

UD: Set of positive integers
Gxy: x equals y

The first sentence is true because each positive integer fails to be equal to all positive integers; and the second sentence is true because every positive integer is equal to itself. Thus both members of the set are true on at least one interpretation.

7. Suppose that **P** and **Q** are quantificationally equivalent. Then on every interpretation **P** and **Q** have the same truth-value. Thus the biconditional **P** ≡ **Q** is true on every interpretation (since a biconditional is true when its immediate components have the same truth-value); hence it is quantificationally true.

Suppose that **P** ≡ **Q** is quantificationally true. Therefore it is true on every interpretation. Then **P** and **Q** have the same truth-value on every interpretation (since a biconditional is true only if its immediate components have the same truth-value) and are quantificationally equivalent.

Section 8.4E

1.a. The set members are true and '(∃x)(Hx & Fx)' false on the following interpretation:

UD: Set of positive integers
Fx: x is evenly divisible by 2
Hx: x is odd
Gx: x is greater than or equal to 1

Every positive integer that is evenly divisible by 2 is greater than or equal to 1, every odd positive integer is greater than or equal to 1, but no positive integer is both evenly divisible by 2 and odd.

c. The set member is true and 'Fa' is false on the following interpretation:

UD: Set of positive integers
Fx: x is even
a: the number 1

At least one positive integer is even, but the number 1 is not even.

e. The set members are true and '(∃x)Bx' is false on the following interpretation:

UD: Set of positive integers
Bx: x is negative
Cx: x is prime

Every positive integer is trivially such that if it is negative then it is prime, for no positive integer is negative; and at least one positive integer is prime. But no positive integer is negative.

g. The set member is true and ‘(∀x) ~ Lxx’ is false on the following interpretation:

UD: Set of positive integers
Lxy: x is greater than or equal to y

Every positive integer x is such that for some positive integer y, x is not greater than or equal to y. But it is false that every positive integer is not greater than or equal to itself.

2.a. The premises are true and the conclusion false on the following interpretation:

UD: Set of positive integers
Fx: x is positive
Gx: x is negative
Nx: x equals 0

The first premise is true since its antecedent is false. The second premise is trivially true because no positive integer equals 0. The conclusion is false for no positive integer satisfies the condition of being either not positive or negative.

c. The premises are true and the conclusion false on the following interpretation:

UD: Set of positive integers
Fx: x is prime
Gx: x is even
Hx: x is odd

There is an even prime positive integer (the number 2), and at least one positive integer is odd and prime, but no positive integer is both even and odd.

e. The premises are true and the conclusion false on the following interpretation:

UD: Set of positive integers
Fx: x is negative
Gx: x is odd

The first premise is trivially true, for no positive integer is negative. For the same reason, the second premise is true. But at least one positive integer is odd, and so the conclusion is false.

g. The premises are true and the conclusion false on the following interpretation:

UD: Set of positive integers
Gx: x is prime
Dxy: x equals y

Some positive integer is prime, and every prime number equals itself, but there is no prime number that is equal to every positive integer.

i. The premises are true and the conclusion false on the following interpretation:

UD: Set of positive integers
Fx: x is odd
Gx: x is positive
Hx: x is prime

Every odd positive integer is positive, and every prime positive integer is positive, but not every positive integer is odd or prime.

3.a. A symbolization of the first argument is

$$\frac{(\forall x)Bx}{(\exists x)Bx}$$

To see that this argument is quantificationally valid, assume that '(∀x)Bx' is true on some interpretation. Then every member of the UD is B. Since every UD is nonempty, it follows that there is at least one member that is B. So '(∃x)Bx' is true as well.

A symbolization of the second argument is

$$\frac{(\forall x)(Px \supset Bx)}{(\exists x)(Px \mathbin{\&} Bx)}$$

The premise is true and the conclusion false on the following interpretation:

UD: Set of positive integers
Px: x is negative
Bx: x is prime

c. One symbolization of the first argument is

$$\frac{(\exists x)(\forall y)Lxy}{(\forall y)(\exists x)Lxy}$$

To see that the argument is quantificationally valid, assume that the premise is true on some interpretation. Then some member x of the UD—let's call it a—stands in the relation L to every member of the UD. Thus for each member y of the UD, there is some member—namely, a—that stands in the relation L to y. So the conclusion is true as well.

A symbolization of the second argument is

$$\frac{(\forall x)(\exists y)Lyx}{(\exists y)(\forall x)Lyx}$$

The following interpretation makes the premise true and the conclusion false:

UD: Set of positive integers
Lxy: x is larger than y

For each positive integer, there is a larger one, but no positive integer is the largest.

e. A symbolization of the first argument is

$$\frac{(\exists x)(Tx \,\&\, Sx) \,\&\, (\exists x)(Tx \,\&\, {\sim}\, Hx)}{(\exists x)(Tx \,\&\, (Sx \vee {\sim}\, Hx))}$$

To see that this argument is quantificationally valid, assume that the premise is true on some interpretation. Then at least one member of the UD—let's call it a—is both T and S and at least one member of the UD is both T and not H. a satisfies the condition of being both T and either S or H, and so the conclusion is true as well.

A symbolization of the second argument is

$$\frac{(\forall x)(Tx \supset Sx) \,\&\, {\sim}\, (\exists x)(Tx \,\&\, Hx)}{(\exists x)(Tx \,\&\, (Sx \vee {\sim}\, Hx))}$$

The following interpretation makes the premise true and the conclusion false:

UD: Set of positive integers
Tx: x is negative
Sx: x is odd
Hx: x is prime

Every negative positive integer (there are none) is odd, and there is no positive integer that is negative and prime. But it is false that some positive integer is both negative and either odd or not prime.

g. A symbolization of the first argument is

$$\begin{array}{l}(\forall x)(Ax \supset Cx)\ \&\ (\forall x)(Cx \supset Sx) \\ \hline (\forall x)(Ax \supset Sx)\end{array}$$

To see that the argument is quantificationally valid, assume that the premise is true on some interpretation. Then every member of the UD that is A is also C, and every member of the UD that is C is also S. So if a member of the UD is A, it is C and therefore S as well, which is what the conclusion says.

A symbolization of the second argument is

$$\begin{array}{l}(\forall x)(Sx \supset Cx)\ \&\ (\forall x)(Cx \supset Ax) \\ \hline (\forall x)(Ax \supset Sx)\end{array}$$

The premise is true and the conclusion false on the following interpretation:

UD: Set of positive integers
Ax: x is positive
Cx: x is greater than 1
Sx: x is even

Every even positive integer is greater than 1, and every positive integer that is greater than 1 is positive. But not every positive integer that is positive is even—some positive integers are odd.

4.a. The argument is quantificationally invalid. The premises are true and the conclusion false on the following interpretation:

UD: Set of positive integers
Dx: x is odd
Fx: x is greater than 10
Lx: x is greater than 9

Every odd positive integer that is greater than 9 is greater than 10; at least one odd positive integer is not greater than 10; but it is false that no positive integer is greater than 9.

c. The argument is quantificationally invalid. The premise is true and the conclusion false on the following interpretation:

UD: Set of positive integers
Hx: x is less than 0
Rx: x is less than −1
Sx: x is less than −2

There is at least one positive integer such that it is less than 0 if and only if it is less than both −1 and −2; every positive integer has this property. But there is no positive integer that is either less than 0 and less than −1 or less than 0 and less than −2.

Section 8.5E

1.a. $Ca \supset Daa$
c. $Ba \vee Faa$
e. $Ca \supset (N \supset Ba)$
g. $Ba \supset Ca$
i. $Ca \vee (Daa \vee Ca)$

2. Remember that, in expanding a sentence containing the individual constant ‘g’, we must use that constant.

a. $Dag \; \& \; Dgg$
c. $[Aa \; \& \; (Daa \vee Dba)] \vee [Ab \; \& \; (Dab \vee Dbb)]$
e. $[Ua \supset ((Daa \vee Daa) \vee (Dab \vee Dba))]$
$\& \; [Ub \supset ((Dba \vee Dab) \vee (Dbb \vee Dbb))]$
g. $[Dag \supset ((\sim Ua \; \& \; Daa) \vee (\sim Ug \; \& \; Dag))]$
$\& \; [Dgg \supset ((\sim Ua \; \& \; Dga) \vee (\sim Ug \; \& \; Dgg))]$
i. $\sim (K \vee ((Daa \; \& \; Dab) \vee (Dba \; \& \; Dbb)))$

3. Remember that if any individual constants occur in a sentence, those constants must be used in the expansion of the sentence.

a. $Bb \; \& \; [(Gab \supset \sim Eab) \; \& \; (Gbb \supset \sim Ebb)]$
c. $[(Gaa \supset \sim Eaa) \; \& \; (Gab \supset \sim Eab)]$
$\& \; [(Gba \supset \sim Eba) \; \& \; (Gbb \supset \sim Ebb)]$

e. Impossible! This sentence contains three individual constants, ‘a’, ‘b’, and ‘c’; so it can be expanded only for sets of at least three constants.

g. $[Ba \supset \sim ((Ba \; \& \; Maaa) \vee (Bb \; \& \; Maab))]$
$\& \; [Bb \supset \sim ((Ba \; \& \; Mbaa) \vee (Bb \; \& \; Mbab))]$
i. $[Eaa \equiv \sim ((Maaa \vee Maba) \vee (Mbaa \vee Mbba))]$
$\& \; [Ebb \equiv \sim ((Maab \vee Mabb) \vee (Mbab \vee Mbbb))]$

4.a. $[(Ga \supset Naa) \; \& \; (Gb \supset Nbb)] \; \& \; (Gc \supset Ncc)$
c. $((Na \equiv Ba) \vee (Na \equiv Bb)) \vee (Na \equiv Bc)$

5. The truth-table for an expansion for the set {‘a’} is

Fa	(Fa	&	~ Fa)	↓ ⊃	~ Fa
T	**T**	**F**	**F T**	**T**	**F T**
F	**F**	**F**	**T F**	**T**	**T F**

This truth-table shows that the the sentence

$$((\exists x)Fx \; \& \; (\exists y) \sim Fy) \supset (\forall x) \sim Fx$$

is true on every interpretation with a one-member UD. The truth-table for an expansion for the set {‘a’, ‘b’} is

											↓					
Fa	Fb	[(Fa	∨	Fb)	&	(~	Fa	∨	~	Fb)]	⊃	(~	Fa	&	~	Fb)
T	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**	**T**	**F**	**F**	**T**
T	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**T**	**F**
F	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**T**
F	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**

This truth-table shows that the sentence

$$((\exists x)Fx \,\&\, (\exists y) \sim Fy) \supset (\forall x) \sim Fx$$

is true on at least one interpretation with a two-member UD and false on at least one interpretation with a two-member UD.

6.a. One assignment to its atomic components for which the expansion

$$[Naa \vee (Naa \vee Nan)] \,\&\, [Nnn \vee (Nna \vee Nnn)]$$

is true is

									↓					
Naa	Nan	Nna	Nnn	[Naa	∨	(Naa	∨	Nan)]	&	[Nnn	∨	(Nna	∨	Nnn)]
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**

Using this information, we shall construct an interpretation with a two-member UD such that the relation N holds between each two members of the UD:

UD: The set {1, 2}
Nxy: x is less than, equal to, or greater than y

Every member of the UD is less than, equal to, or greater than both itself and the other member of the UD, and so ‘$(\forall x)(Nxx \vee (\exists y)Nxy)$’ is true on this interpretation.

c. There is only one assignment to its atomic components for which the expansion ‘Saan & Snnn’ is true.

			↓	
Saan	Snnn	Saan	&	Snnn
T	**T**	**T**	**T**	**T**

Using this information, we construct an interpretation with a two-member UD:

UD: The set {1, 2}
Sxyz: x equals y times z
a: 2
n: 1

Because 1 = 1 × 1 and 2 = 2 × 1, '(∀y)Syyn' is true on this interpretation.

7.a.

					↓	
Fa	Ga	(Fa	⊃	Ga)	⊃	Ga
F	**F**	**F**	**T**	**F**	**F**	**F**

c.

Baa	Bab	Bba	Bbb	[(Baa	∨	Bab)	&	(Bba	∨	Bbb)]
T	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**

↓							
⊃	[(Baa	&	Bba)	∨	(Bab	&	Bbb)]
F	**T**	**F**	**F**	**F**	**F**	**F**	**T**

e.

											↓							
Fa	Ga	Fb	Gb	[(Fa	&	Fb)	⊃	(Ga	&	Gb)]	⊃	[(Fa	⊃	Ga)	&	(Fb	⊃	Gb)]
T	**F**	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**F**	**F**	**F**	**T**	**T**

g.

				↓			
Faa	Ga	~	Ga	⊃	(Faa	⊃	Ga)
T	**F**	**T**	**F**	**F**	**T**	**F**	**F**

8.a.

											↓			
Baa	Bab	Bba	Bbb	~	[(Baa	&	Bab)	&	(Bba	&	Bbb)]	≡	(Baa	&	Bbb)
T	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**

c.

Fa	Fb	Ga	Gb	[(Fa	∨	Fb)	&	(Ga	∨	Gb)]
T	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**

↓								
&	~	[(Fa	&	Ga)	∨	(Fb	&	Gb)]
T	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**

e.

Fa	Ga	(Fa	⊃	Ga)	↓ &	(Ga	⊃	~	Fa)
F	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**

g.

Ba	Ha	(Ba	≡	Ha)	↓ ⊃	(Ba	&	Ha)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**

i. Sneaky. This one can't be done because, as pointed out in Section 8.2, the sentence is false on all interpretations with finite UDs.

9.a.

Fa	Fb	Ga	Gb	((Fa	&	Ga)	∨	(Fb	&	Gb))
T	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**F**	**F**

↓ ⊃	(~	(Fa	∨	Ga)	∨	~	(Fb	∨	Gb))
T	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**F**

Fa	Fb	Ga	Gb	((Fa	&	Ga)	∨	(Fb	&	Gb))
T	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**T**

↓ ⊃	(~	(Fa	∨	Ga)	∨	~	(Fb	∨	Gb))
F	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**

c.

Bnn	Bnn	↓ ⊃	~	Bnn
F	**F**	**T**	**T**	**F**

Bnn	Bnn	↓ ⊃	~	Bnn
T	**T**	**F**	**F**	**T**

e.

Naa	(Naa	∨	Naa)	↓ ⊃	Naa
T	**T**	**T**	**T**	**T**	**T**

Naa	Nab	Nba	Nbb	[[(Naa	∨	Naa)	⊃	Naa]
T	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**

						↓					
&	[(Nba	∨	Nab)	⊃	Nbb]]	&	[[(Nab	∨	Nba)	⊃	Naa]
F	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**

&	[(Nbb	∨	Nbb)	⊃	Nbb]]
T	**F**	**F**	**F**	**T**	**F**

g.

					↓			
Ca	Da	(Ca	∨	Da)	≡	(Ca	&	Da)
T	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**

					↓			
Ca	Da	(Ca	∨	Da)	≡	(Ca	&	Da)
T	**F**	**T**	**T**	**F**	**F**	**T**	**F**	**F**

11. The expanded sentence ‘Ga & ~ Ga’ is a truth-functional compound. It is false on every truth-value assignment, so it is quantificationally false. But the fact that this sentence is quantificationally false only shows that ‘(∃y)Gy & (∃y) ~ Gy’ is not true on any interpretation that has a one-member UD—for it is an expansion using only one constant. The sentence is in fact not quantificationally false, for it is true on some interpretations with larger universes of discourse. We may expand the sentence for the set {‘a’, ‘b’} to show this:

					↓						
Ga	Gb	(Ga	∨	Gb)	&	(~	Ga	∨	~	Gb)
T	**F**	**T**	**T**	**F**	**T**	**F**	**T**	**T**	**T**	**F**

12.a.

						↓					↓			
Fa	Fb	Ga	(Fa	∨	Fb)	⊃	Ga	(Fa	⊃	Ga)	∨	(Fb	⊃	Ga)
T	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**

c.

							↓							↓			
Fa	Fb	Ga	Gb	(Fa	&	Fb)	∨	(Ga	&	Gb)	(Fa	∨	Ga)	&	(Fb	∨	Gb)
T	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**

e.

							↓							↓			
Fa	Fb	Ga	Gb	(Fa	≡	Ga)	&	(Fb	≡	Gb)	(Fa	∨	Fb)	≡	(Ga	∨	Gb)
T	**F**	**F**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**T**	**T**	**F**	**T**	**F**	**T**	**T**

g.

											↓					
Ba	Bb	Daa	Dab	Dba	Dbb	(Ba	&	(Daa	&	Dba))	∨	(Bb	&	(Dab	&	Dbb))
F	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**

					↓					
(Ba	⊃	(Daa	&	Dba))	&	(Bb	⊃	(Dab	&	Dbb))
F	**T**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**T**

i.

														↓							
Fa	Fb	Kaa	Kab	Kba	Kbb	((Fa	⊃	Kaa)	∨	(Fa	⊃	Kba))	&	((Fb	⊃	Kab)	∨	(Fb	⊃	Kbb))
T	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**

							↓							
((Fa	⊃	Kaa)	∨	(Fa	⊃	Kba))	∨	((Fb	⊃	Kab)	∨	(Fb	⊃	Kbb))
T	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**T**	**T**

13.a.

					↓			↓		↓							
Ba	Bb	Ca	Cb	Ba	∨	Bb	Ca	∨	Cb	~	[(Ba	∨	Ca)	&	(Bb	∨	Cb)]
T	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**F**

c.

					↓			↓			↓		
Fa	Ga	Ma	Na	Fa	⊃	Ga	Na	⊃	Ma	Ga	⊃	~	Ma
F	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**T**	**F**

e.

				↓					↓		
Caa	Ma	Na	Na	⊃	(Ma	&	Caa)	Ma	⊃	~	Caa
T	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**T**	**F**	**T**

g.

				↓							
Ma	Mb	Na	Nb	~	[(Na	⊃	Ma)	&	(Nb	⊃	Mb)]
F	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**T**	**T**	**T**

↓									
~	[~	(Na	⊃	Ma)	&	~	(Nb	⊃	Mb)]
T	**T**	**T**	**F**	**F**	**F**	**F**	**T**	**T**	**T**

i.

		↓	↓			↓	
Faa	Gaa	Faa	~	Gaa	Faa	∨	Gaa
T	**F**	**T**	**T**	**F**	**T**	**T**	**F**

15.a.

Fa	Ga	Na	(Fa	⊃	Ga)	↓ ⊃	Na	Na	↓ ⊃	Ga	~	Fa	↓ ∨	Ga
T	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**

c.

Fa	Fb	Ga	Gb	Ha	Hb	(Fa	&	Ga)	↓ ∨	(Fb	&	Gb)
T	**T**	**T**	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**T**	**F**	**F**

(Fa	&	Ha)	↓ ∨	(Fb	&	Hb)	(Ga	&	Ha)	↓ ∨	(Gb	&	Hb)
T	**F**	**F**	**T**	**T**	**T**	**T**	**T**	**F**	**F**	**F**	**F**	**F**	**T**

e.

Fa	Ga	Fa	↓ ⊃	Ga	↓ ~	Fa	↓ ~	Ga
F	**T**	**F**	**T**	**T**	**T**	**F**	**F**	**T**

g.

Daa	Dab	Dba	Dbb	Ga	Gb	Ga	↓ ∨	Gb	(Ga	⊃	Daa)	↓ &	(Gb	⊃	Dbb)
T	**F**	**F**	**T**	**F**	**T**	**F**	**T**	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**

[(Ga	&	Daa)	&	(Ga	&	Dab)]	↓ ∨	[(Gb	&	Dba)	&	(Gb	&	Dbb)]
F	**F**	**T**	**F**	**F**	**F**	**F**	**F**	**T**	**F**	**F**	**F**	**T**	**T**	**T**

i.

Fa	Ga	Ha	Fa	↓ ⊃	Ga	Ha	↓ ⊃	Ga	Fa	↓ ∨	Ha
F	**F**	**F**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**F**

Section 8.6E

1.a. **F**
c. **T**
e. **T**
g. **F**
i. **F**

2.a. The sentence is false on the following interpretation:

UD: Set of positive integers

There is no positive integer that is identical to every positive integer.

c. The sentence is false on the following interpretation:

UD: The set {1, 2, 3}

It is not true that for any three members of the UD, at least two are identical.

e. The sentence is false on the following interpretation:

UD: The set {1}
Gxy: x is greater than y

It is not true that there is a pair of members of the UD such that either the members of the pair are not identical or one member is greater than the other. The only pair of members of the UD consists of 1 and 1.

3.a. Consider any interpretation and any members x, y, and z of its UD. If x and y are not the same member or if y and z are not the same member, then these members do not satisfy the condition specified by '(x = y & y = z)', and so they do satisfy '[(x = y & y = z) ⊃ x = z]'. On the other hand, if x and y are the same and y and z are the same, then x and z must be the same, satisfying the consequent 'x = z'. In this case as well, then, x, y, and z satisfy '[(x = y & y = z) ⊃ x = z]'. Therefore the universal claim is true on every interpretation.

c. Consider any interpretation and any members x and y of its UD. If x and y are not the same, they do not satisfy 'x = y' and so do satisfy '[x = y ⊃ (Gxy ≡ Gyx)]'. If x and y are the same, and hence satisfy 'x = y', they must satisfy '(Gxy ≡ Gyx)' as well—the pair consisting of the one object and itself is either in the extension or not. Therefore the universal claim must be true on every interpretation.

4.a. The first sentence is true and the second false on the following interpretation:

UD: Set of positive integers

Every positive integer is identical to at least one positive integer (itself), but not even one positive integer is identical to every positive integer.

c. The first sentence is false and the second is true on the following interpretation:

UD: Set of positive integers
a: 1
b: 1
c: 2
d: 3

5.a. The sentences are all true on the following interpretation:

UD: Set of positive integers
a: 1
b: 1
c: 1
d: 2

c. The sentences are all true on the following interpretation:

UD: Set of positive integers

The first sentence is true because there are at least two positive integers. The second sentence is true because for any positive integer x, we can find a pair of positive integers z and w such that either x is identical to z or x is identical to w—just let one of the pair be x itself.

6.a. The following interpretation shows that the entailment does not hold:

UD: The set {1, 2}

It is true that for any x, y, and z in the UD, at least two of x, y, and z must be identical. But it is not true that for any x and y in the UD, x and y must be identical.

c. The following interpretation shows that the entailment does not hold:

UD: The set {1, 2}
Gxy: x is greater than or equal to y

At least one member of the UD (the number 2) is greater than or equal to every member of the UD, and at least one member of the UD (the number 1) is not greater than or equal to any member of the UD other than itself. But no member of the UD is not greater than or equal to itself.

7.a. The argument can be symbolized as

$$\frac{(\forall x)[Mx \supset (\exists y)(\sim y = x \ \&\ Lxy)] \ \&\ (\forall x)[Mx \supset (\forall y)(Pxy \supset Lxy)]}{(\forall x)(Mx \supset \sim Pxx)}$$

The argument is quantificationally invalid, as the following interpretation shows:

UD: Set of positive integers
Mx: x is odd
Lxy: x is less than or equal to y
Pxy: x squared equals y

For every odd positive integer, there is at least one other positive integer that it is less than or equal to, and every odd positive integer is such that it is less than or equal to its square(s). However, the conclusion, which says that no odd positive integer is its own square, is false because the square of 1 is 1.

c. The argument can be symbolized as

$$\begin{array}{l}
(\forall x)\ [(Fx\ \&\ (\exists y)(Pxy\ \&\ Lxy)) \supset Lxx] \\
\hline
(\forall x)\ [Fx \supset (\exists y)(\exists z)((Lxy\ \&\ Lxz)\ \&\ {\sim}\, y = z)]
\end{array}$$

The argument is quantificationally invalid, as the following interpretation shows:

UD: Set of positive integers
Fx: x is odd
Lxy: x is greater than y
Pxy: x is less than y

Trivially, every odd positive integer that is both less than and greater than some positive integer (there are none) is less than itself. But not all odd positive integers are greater than at least two positive integers—the number 1 is not.

e. The argument may be symbolized as

$$\begin{array}{l}
(\forall x) \sim (\exists y)(\exists z)(\exists w)([[Pyz\ \&\ Pzx)\ \&\ Pwx] \\
\quad \&\ [({\sim}\, y = z\ \&\ {\sim}\, z = w)\ \&\ {\sim}\, w = y]] \\
\quad \&\ (\forall x_1)[Px_1x \supset ((x_1 = y \vee x_1 = z) \vee x_1 = w)]) \\
(\forall x)(\exists y)(\exists z)[(Pyx\ \&\ Pzx)\ \&\ {\sim}\, y = z)] \\
\hline
(\forall x)(\exists y)(\exists z)[((Pyx\ \&\ Pzx)\ \&\ {\sim}\, y = z)\ \&\ (\forall w)(Pwx \supset (w = y \vee w = z))]
\end{array}$$

The argument is quantificationally invalid, as the following interpretation shows:

UD: Set of positive integers
Pxy: x is greater than y

No positive integer is less than exactly three positive integers (for any positive integer, there are infinitely many positive integers that are greater). Every positive integer is less than at least two positive integers. But no positive integer is less than exactly two positive integers.

8.a.

	↓
a = a	~ a = a
T	F T

									↓					
a = a	a = b	b = a	b = b	(~	a = a	∨	~	b = a)	∨	(~	a = b	∨	~	b = b)
T	**F**	**F**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**

c.

					↓		↓
a = a	Gaa	(Gaa	∨	Gaa)	∨	a = a	Gaa
T	**F**	**F**	**F**	**F**	**T**	**T**	**F**

e.

					↓							↓					
a = a	a = b	b = a	b = b	a = a	&	b = b	(~	a = a	∨	~	a = b)	∨	(~	b = a	∨	~	b = b)
T	**F**	**F**	**T**	**T**	**T**	**T**	**F**	**T**	**T**	**T**	**F**	**T**	**T**	**F**	**T**	**F**	**T**

					↓																		
(~	a = a	&	~	a = b)	∨	(~	b = a	&	~	b = b)													
F	**T**	**F**	**T**	**F**	**F**	**T**	**F**	**F**	**F**	**T**													

9.a. True. Every positive integer is less than its successor.

c. True. For any positive integer x, there is a positive integer that equals 2x.

e. False. The sum of any even integer and any odd integer is odd, not even.

g. True. For any positive integer x there is a positive integer z that satisfies the first disjunct, namely, x squared plus z is even.

10.a. The sentence is false on the following interpretation:

UD: Set of positive integers
Px: x is odd
f(x): the successor of x

It is false that a positive integer with an odd successor is itself odd.

c. The sentence is false on the following interpretation:

UD: Set of positive integers
g(x): the successor of x

There is no positive integer that is the successor of every positive integer.

e. The sentence is false on the following interpretation:

UD: Set of positive integers
f(x): x squared

Since $1 = 1^2$, not all positive integers fail to be equal to their squares.

11.a. The sentence is true on an interpretation if and only if every member x of the UD satisfies '$(\exists y)\ y = f(f(x))$', and that is the case if and only if for every member x of the UD, there is a member y such that y is identical to $f(f(x))$. Since f is a function that is defined for every member of the UD, there must be a member that is identical to $f(x)$, and hence there must also be a member that is identical to $f(f(x))$. Hence the sentence is true on every interpretation.

c. Assume that the antecedent is true on some interpretation. By the first conjunct, it must be the case that every member x of the UD stands in the relation H to $f(x)$, and also that every member $f(x)$ stands in the relation H to $f(f(x))$. By the second conjunct it follows that every member x of the UD therefore stands in the relation H to $f(f(x))$. The consequent must therefore be true as well. Since the consequent is true on every interpretation on which the antecedent is true, the sentence is quantificationally true.

12.a. The first sentence is true and the second false on the following interpretation:

UD: Set of positive integers
Lxyz: x plus y equals z
$f(x)$: the successor of x
a: 1
b: 2

The sum of 1 and 2 is 3, the successor of 2; but the sum of 1 and 3 is not 2.

c. The first sentence is true and the second false on the following interpretation:

UD: Set of positive integers
$f(x)$: x squared
$g(x)$: the successor of x

For any positive integer x, there is a positive integer that is equal to the square of the successor of x; but there is no positive integer that is equal to its own successor squared.

13.a. The members of the set are all true on the following interpretation:

UD: Set of positive integers
$f(x)$: x squared
a: 1
b: 1
c: 1

The number 1 equals itself squared, which is what each of the three sentences in the set say on this interpretation.

c. The members of the set are all true on the following interpretation:

UD: Set of positive integers
f(x): the smallest odd integer that is less than or equal to x

There is a positive integer, namely 1, that is the smallest odd integer less than or equal to any positive integer, and there is at least one positive integer, for example 2, that fails to be the smallest odd integer less than or equal to even one positive integer.

14.a. The argument is quantificationally invalid, as the following interpretation shows:

UD: Set of positive integers
Fx: x is odd
g(x): the successor of x

The premise, which says that every positive integer is such that either it or its successor is odd, is true on this interpretation. The conclusion, which says that every positive integer is such that either it or the successor of its successor is odd, is false—no even positive integer satisfies this condition.

c. The argument is quantificationally invalid, as the following interpretation shows:

UD: Set of positive integers
Lxyz: x plus y equals z
f(x): the successor of x

The premise is true on this interpretation: every positive integer is such that its successor plus some positive integer equals a positive integer. The conclusion is false: there is no positive integer such that the sum of x and any integer's successor equals any integer's successor.

e. The argument is quantificationally valid. If the premise is true on an intepretation, then every member x of the UD that is a value of the function g and that is B is such that nothing stands in the relation H to x. If the antecedent of the conclusion is true, then a is a value of the function g (for the argument b), and is such that something stands in the relation H to a. It follows from the premise that the consequent of the conclusion must be true as well, i.e., a cannot be B. So the conclusion is true on any interpretation on which the premise is true.

15.a.

				↓		↓
$a = g(a)$	Fa	$Fg(a)$	Fa	$\vee$	$Fg(a)$	$a = g(a)$
T	**T**	**T**	**T**	**T**	**T**	**T**

				↓		↓
$a = g(a)$	Fa	$Fg(a)$	Fa	$\vee$	$Fg(a)$	$a = g(a)$
T	**F**	**F**	**F**	**F**	**F**	**T**

c.

$a = f(a)$	$a = f(b)$	$a = f(f(a))$	$a = f(f(b))$	$b = f(a)$	$b = f(b)$	$b = f(f(a))$
F	**T**	**T**	**F**	**T**	**F**	**F**

			↓				↓	
$b = f(f(b))$	$\sim$	$a = f(a)$	$\&$	$\sim$	$b = f(b)$	$a = f(a)$	$\vee$	$b = f(a)$
T	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**

	↓			↓			↓	
$a = f(b)$	$\vee$	$b = f(b)$	$a = f(f(a))$	$\vee$	$b = f(f(a))$	$a = f(f(b))$	$\vee$	$b = f(f(b))$
T	**T**	**F**	**T**	**T**	**F**	**F**	**T**	**T**

Section 8.7E

1.a. Let **d** be a variable assignment for this interpretation. **d** satisfies the antecedent ‘~ (∀x)Ex’ just in case it fails to satisfy ‘(∀x)Ex’. **d** fails to satisfy ‘(∀x)Ex’ just in case there is at least one member **u** of the UD such that **d**[**u**/x] fails to satisfy ‘Ex’. The number 1 is such a member: **d**[1/x] fails to satisfy ‘Ex’ because ⟨**d**[1/x](x)⟩, which is ⟨1⟩, is not a member of **I**(E), the set of 1-tuples of even positive integers. So **d** satisfies ‘~ (∀x)Ex’.

d satisfies the consequent ‘(∃y) Lyo’ when there is at least one member **u** of the UD such that **d**[**u**/y] satisfies ‘Lyo’, that is, just in case there is at least one member **u** such that ⟨**d**[**u**/y](y), **I** (o)⟩, which is ⟨**u**, 1⟩, is in **I**(L). There is no such member, for there is no positive integer that is less than 1. Therefore **d** does not satisfy ‘(∃y)Lyo’ and consequently **d** does not satisfy the conditional ‘~ (∀x)Ex ⊃ (∃y)Lyo’. The sentence is false on this interpretation.

c. Let **d** be a variable assignment for this interpretation. **d** satisfies ‘(∃x)(Ko ∨ Ex)’ just in case there is some member **u** of the UD such that **d**[**u**/x] satisfies ‘Ko ∨ Ex’. There is such a member—take 2 as an example. **d**[2/x] satisfies ‘Ko ∨ Ex’ because **d**[2/x] satisfies the second disjunct. **d**[2/x] satisfies ‘Ex’ because ⟨**d**[2/x](x)⟩, which is ⟨2⟩, is a member of **I**(E)—2 is even. Therefore **d** satisfies ‘(∃x)(Ko ∨ Ex)’. The sentence is true on this interpretation.

e. Let **d** be a variable assignment for this interpretation. **d** satisfies ‘(Ko ≡ (∀x)Ex) ⊃ (∃y)(∃z)Lyz’ if and only if either **d** fails to satisfy the antecedent or **d** does satisfy the consequent. **d** satisfies the antecedent because it fails to satisfy both ‘Ko’ (no satisfaction assignment satisfies this formula) and ‘(∀x) Ex’. **d** does not satisfy the latter because not every member **u** of the UD is such that **d**[**u**/x] satisfies ‘Ex’—no odd number is in the extension of ‘E’.

d also satisfies the consequent '(∃y)(∃z)Lyz' because, for example, **d**[1/y] satisfies '(∃z)Lyz'. The latter is the case because, for example, **d**[1/y, 2/z] satisfies 'Lyz'; ⟨1, 2⟩ is in the extension of 'L'. The sentence is true on this interpretation.

2.a. Let **d** be a variable assignment for this interpretation. **d** satisfies '(∃x)(Ex ⊃ (∀y)Ey)' just in case there is at least one member **u** of the UD such that **d**[**u**/x] satisfies 'Ex ⊃ (∀y)Ey'. There is such a member; take 1 as an example. **d**[1/x] satisfies 'Ex ⊃ (∀y)Ey' because it fails to satisfy 'Ex'. **d**[1/x] fails to satisfy 'Ex' because ⟨**d**[1/x](x)⟩, which is ⟨1⟩, is not a member of **I**(E)—1 is not even. So **d** satisfies '(∃x)(Ex ⊃ (∀y)Ey)'. The sentence is true on this interpretation.

c. Let **d** be a variable assignment for this interpretation. **d** satisfies '(∀x)(Tx ⊃ (∃y)Gyx)' just in case every member **u** of the UD is such that **d**[**u**/x] satisfies 'Tx ⊃ (∃y)Gyx', that is, just in case both **d**[1/x] and **d**[3/x] satisfy 'Tx ⊃ (∃y)Gyx'. **d**[1/x] satisfies 'Tx ⊃ (∃y)Gyx' because it satisfies '(∃y)Gyx'. **d**[1/x] satisfies '(∃y)Gyx' because there is at least one member **u** of the UD such that **d**[1/x, **u**/y] satisfies 'Gyx'—3 is such a member. **d**[1/x, 3/y] satisfies 'Gyx' because ⟨**d**[1/x, 3/y](y), **d**[1/x, 3/y](x)⟩, which is ⟨3, 1⟩, is a member of **I**(G)—3 is greater than 1.

d[3/x] satisfies 'Tx ⊃ (∃y)Gyx' because **d**[3/x] does not satisfy 'Tx'. **d**[3/x] does not satisfy 'Tx' because ⟨**d**[3/x](x)⟩, which is ⟨3⟩, is not a member of **I**(**T**)—3 is not less than 2. So both **d**[1/x] and **d**[3/x] satisfy 'Tx ⊃ (∃y)Gyx' and therefore **d** satisfies '(∀x)(Tx ⊃ (∃y)Gyx)'. The sentence is true on this interpretation.

e. Let **d** be a variable assignment for this interpretation. **d** satisfies this sentence just in case for every member **u** of the UD, **d**[**u**/x] satisfies '(∀y)Gxy ∨ (∃y)Gxy'. However, the number 1 is *not* such that **d**[1/x] satisfies the formula. **d**[1/x] does not satisfy '(∀y)Gxy', because there is not even one member **u** of the UD such that **d**[1/x, **u**/y] satisfies 'Gxy'—no 2-tuple ⟨1, **u**⟩ is in the extension of 'G'. **d**[1/x] also does not satisfy '(∃y)Gxy', for the same reason. Because **d**[1/x] does not satisfy '(∀y)Gxy ∨ (∃y)Gxy', **d** does not satisfy the universally quantified sentence. The sentence is false on this interpretation.

3.a. Let **d** be a variable assignment for this interpretation. **d** satisfies 'Mooo ≡ Pooo' just in case either **d** satisfies both 'Mooo' and 'Pooo' or **d** satisfies neither of 'Mooo' and 'Pooo'. **d** does not satisfy 'Mooo' because ⟨**I**(o), **I**(o), **I**(o)⟩, which is ⟨1, 1, 1⟩, is not a member of **I**(M)—1 − 1 ≠ 1. **d** does not satisfy 'Pooo' because ⟨**I**(o), **I**(o), **I**(o)⟩, which again is ⟨1, 1, 1⟩, is not a member of **I**(P)—1 + 1 ≠ 1. So **d** satisfies neither immediate component and therefore does satisfy 'Mooo ≡ Pooo'. The sentence is true on this interpretation.

c. Let **d** be a variable assignment for this interpretation. **d** satisfies '(∀x)(∀y)(∀z)(Mxyz ≡ Pxyz)' just in case every member **u** of the UD is such that **d**[**u**/x] satisfies '(∀y)(∀z)(Mxyz ≡ Pxyz)'. **d**[**u**/x] satisfies '(∀y)(∀z)(Mxyz ≡ Pxyz)' just in case every member $\mathbf{u}_1$ of the UD is such that **d**[**u**/x, $\mathbf{u}_1$/y] satisfies '(∀z)(Mxyz ≡ Pxyz)'. **d**[**u**/x, $\mathbf{u}_1$/y] satisfies '(∀z)(Mxyz ≡ Pxyz)' just in case every member $\mathbf{u}_2$ of the UD is such that **d**[**u**/x, $\mathbf{u}_1$/y, $\mathbf{u}_2$/z] satisfies 'Mxyz ≡ Pxyz'. So **d** satisfies '(∀x)(∀y)(∀z)(Mxyz ≡ Pxyz) just in case for any members **u**, $\mathbf{u}_1$, and $\mathbf{u}_2$ of the UD, **d**[**u**/x, $\mathbf{u}_1$/y, $\mathbf{u}_2$/z] satisfies 'Mxyz ≡ Pxyz'. But this is not the case. For example, **d**[1/x, 2/y, 3/z] does not satisfy 'Mxyz', because ⟨**d**[1/x, 2/y, 3/z](x), **d**[1/x, 2/y, 3/z](y), **d**[1/x, 2/y, 3/z](z)⟩, which is ⟨1, 2, 3⟩, is not a member of **I**(M)—$1 - 2 \neq 3$. On the other hand, **d**[1/x, 2/y, 3/z] does satisfy 'Pxyz', because ⟨**d**[1/x, 2/y, 3/z](x), **d**[1/x, 2/y, 3/z](y), **d**[1/x, 2/y, 3/z](z)⟩, which again is ⟨1, 2, 3⟩, is a member of **I**(P)—$1 + 2 = 3$. The assignment **d**[1/x, 2/y, 3/z] therefore does not satisfy 'Mxyz ≡ Pxyz', and so **d** does not satisfy '(∀x)(∀y)(∀z)(Mxyz ≡ Pxyz)'. The sentence is false on this interpretation.

e. Let **d** be a variable assignment for this interpretation. **d** satisfies this sentence if and only if for every member **u** of the UD, **d**[**u**/y] satisfies '(∃z)(Pyoz ⊃ Pooo)'. The latter is the case for a member **u** of the UD if and only if there is a member $\mathbf{u}_1$ of the UD such that **d**[**u**/y, $\mathbf{u}_1$/z] satisfies 'Pyoz ⊃ Pooo'. No variable assignment can satisfy 'Pooo', for ⟨1, 1, 1⟩ is not in the extension of 'P'. But for any member **u** of the UD we can find a member $\mathbf{u}_1$ such that ⟨**u**, 1, $\mathbf{u}_1$⟩, is not in the extension of 'P'; pick any number other than the number that is the successor of **u**. The sentence is true on this interpretation.

5. We shall show that the sentence is true on every interpretation. Let **I** be any interpretation. '(∀x)((∀y)Fy ⊃ Fx)' is true on **I** if and only if every variable assignment satisfies the sentence. A variable assignment **d** satisfies '(∀x)((∀y)Fy ⊃ Fx)' if and only if every member **u** of the UD is such that **d**[**u**/x] satisfies '(∀y)Fy ⊃ Fx'. Consider any member **u** of the UD. If ⟨**u**⟩ is a member of **I**(**F**), then **d**[**u**/x] satisfies 'Fx' and hence also satisfies '(∀y)Fy ⊃ Fx'. If ⟨**u**⟩ is not a member of **I**(F), then **d**[**u**/x] does not satisfy '(∀y)Fy'. This is because **u** is such that **d**[**u**/x, **u**/y] does not satisfy 'Fy'—⟨**d**[**u**/x, **u**/y](y)⟩, which is ⟨**u**⟩, is not a member of **I**(F). So if ⟨**u**⟩ is not a member of **I**(F), then **d**[**u**/x] satisfies '(∀y)Fy ⊃ Fx' because it fails to satisfy the antecedent. Each member **u** of the UD is such that either ⟨**u**⟩ is a member of **I**(F) or it isn't, so each member **u** of the UD is such that **d**[**u**/x] satisfies '(∀y)Fy ⊃ Fx'. Therefore d must satisfy '(∀x)((∀y)Fy ⊃ Fx)'. The sentence is true on every interpretation.

7. Assume that 'Fa' is true on an interpretation. Then every variable assignment for this interpretation satisfies 'Fa'. So we know that ⟨**I**(a)⟩ is in the extension of 'F'. We shall now show that every variable assignment also satisfies '(∃x)Fx'. Let **d** be any such assignment. **d** satisfies '(∃x)Fx' if and only if there is some member **u** of the UD such that **d**[**u**/x] satisfies 'Fx'. We know that there is such a member, namely, **I**(a). **d**[**I**(a)/x] satisfies 'Fx' because ⟨**I**(a)⟩ is in the extension of 'F'. Therefore '(∃x)Fx' is true on the interpretation as well.

9.a. Let **d** be a variable assignment for this interpretation. Then **d** satisfies '(∀x)(∀y)[~ x = y ⊃ (Ex ⊃ Gxy)]' if and only if for every positive integer **u**, **d**[**u**/x] satisfies '(∀y)[~ x = y ⊃ (Ex ⊃ Gxy)]'. This will be the case if and only if for every pair of positive integers **u** and $\mathbf{u}_1$, **d**[**u**/x, $\mathbf{u}_1$/y] satisfies '~ x = y ⊃ (Ex ⊃ Gxy)'. But **d**[2/x, 3/y], for example, does not satisfy the open sentence. **d**[2/x, 3/y] does satisfy '~ x = y', for 2 and 3 are distinct members of the UD. **d**[2/x, 3/y] does not satisfy 'Ex ⊃ Gxy', for it satisfies the antecedent and fails to satisfy the consequent. **d**[2/x, 3/y] satisfies 'Ex' because ⟨**d**[2/x, 3/y](x)⟩, which is ⟨2⟩, is a member of **I** (E). **d**[2/x, 3/y] fails to satisfy 'Gxy' because ⟨**d**[2/x, 3/y](x), **d**[2/x, 3/y](y)⟩, which is ⟨2, 3⟩, is not a member of **I**(G)—2 is not greater than 3. We conclude that '(∀x)(∀y)[~ x = y ⊃ (Ex ⊃ Gxy)]' is false on this interpretation.

c. Let **d** be a variable assignment for this interpretation. Then **d** satisfies the sentence if and only if for every member **u** of the UD, **d**[**u**/x] satisfies 'Ex ⊃ (∃y)(~ x = y & ~ Gxy). Every odd positive integer **u** is such that **d**[**u**/x] satisfies the formula because every odd positive integer **u** is such that **d**[**u**/x] fails to satisfy 'Ex'. Every even positive integer **u** is such that **d**[**u**/x] satisfies the formula because every positive integer (odd or even) satisfies the consequent, '(∃y)(~ x = y & ~ Gxy)'. For every positive integer **u** there is a positive integer $\mathbf{u}_1$ such that **d**[**u**/x, $\mathbf{u}_1$/y] satisfies '~ x = y & ~ Gxy': Let $\mathbf{u}_1$ be any integer that is greater than **u**. In this case, **d**[**u**/x, $\mathbf{u}_1$/y] satisfies '~ x = y' becasue **u** and $\mathbf{u}_1$ are not identical, and the variant also satisfies '~ Gxy' because ⟨**u**, $\mathbf{u}_1$⟩ is not in the extension of 'G'. The sentence is therefore true on this interpretation.

10.a. A sentence of the form (∀**x**)**x** = **x** is true on an interpretation **I** if and only if every variable assignment satisfies the sentence on **I**. A variable assignment **d** satisfies (∀**x**)**x** = **x** if and only if for every member **u** of the UD, **d**[**u**/**x**] satisfies **x** = **x**—and this is the case if and only for every member **u** of the UD, **d**[**u**/**x**](**x**) is identical to **d**[**u**/**x**](**x**). Trivially, this is so. Therefore (∀**x**)**x** = **x** is satisfied by every variable assignment on every interpretation; it is quantificationally true.

11.a. Let **d** be a variable assignment for this interpretation. **d** satisfies the universally quantified sentence just in case every member **u** of the UD is such that **d**[**u**/x] satisfies 'O*h*(x) ⊃ O*g*(x,x)'. A member **u** of the UD satisfies the antecedent 'O*h*(x)' just in case the member **u**′ of the UD such that ⟨**u**, **u**′⟩ is a member of **I**(*h*) is itself a member of **I**(O). This will be the case if **u** is odd, since **u**′, its square, will also be odd. But now we note that for every (odd or even) positive integer **u**, **d**[**u**/x] will fail to satisfy the consequent 'O*g*(x,x)'. This is because the member **u**′ of the UD such that ⟨**u**, **u**, **u**′⟩ is a member of **I**(*g*) must be odd, but no positive integer **u**′ that is double a positive integer **u** can be odd. So every odd positive integer **u** is such that **d**[**u**/x] fails to satisfy 'O*h*(x) ⊃ O*g*(x,x)', so **d** fails to satisfy the universally quantified sentence and hence the sentence is false.

c. Let **d** be a variable assignment for this interpretation. **d** satisfies the sentence just in case for at least one pair of members **u** and **u**′ of the UD, **d**[**u**/x,**u**′/y] satisfies ‘Ox & x = *h*(y)’. This will be the case if for at least one pair of members **u** and **u**′ of the UD, ⟨**u**⟩ is a member of **I**(O) and ⟨**u**′, **u**⟩ is a member of **I**(*h*), i.e., **u** is odd and **u** is the square of **u**′. The positive integers 9 and 3 satisfy this condition, so **d**[9/x,3/y] satisfies ‘Ox & x = *h*(y)’, **d** satisfies the existentially quantified sentence, and hence the sentence is true on this interpretation.

12.a. A sentence of the form (∀**x**)(∃**y**)**y** = ***f***(**x**) is quantificationally true just in case it is satisfied by every variable assignment **d** on every interpretation **I**. A variable assignment **d** will satisfy the sentence just in case for every member **u** of the UD there is a member **u**′ of the UD such that **d**[**u**/x, **u**′/y] satisfies **y** = ***f***(**x**). The latter holds just in case for every member **u** of the UD there is a member **u**′ of the UD such that ⟨**u**, **u**′⟩ is a member of **I**(***f***). And this will be the case because of our requirement that **I**(***f***) must always be a function on the UD.

CHAPTER NINE

Section 9.1E

a.	1.	(∃x)Fx✓	SM
	2.	(∃x) ~ Fx✓	SM
	3.	Fa	1 ∃D
	4.	~ Fb	2 ∃D
		o	

The tree has a completed open branch.

c.	1.	(∃x)(Fx & ~ Gx)✓		SM
	2.	(∀x)(Fx ⊃ Gx)		SM
	3.	Fa & ~ Ga✓		1 ∃D
	4.	Fa		3 &D
	5.	~ Ga		3 &D
	6.	Fa ⊃ Ga✓		2 ∀D
	7.	~ Fa	Ga	6 ⊃D
		×	×	

The tree is closed.

e.	1.	~ (∀x)(Fx ⊃ Gx)✓	SM
	2.	~ (∃x)Fx✓	SM
	3.	~ (∃x)Gx✓	SM
	4.	(∃x) ~ (Fx ⊃ Gx)✓	1 ~ ∀D
	5.	(∀x) ~ Fx	2 ~ ∃D
	6.	(∀x) ~ Gx	3 ~ ∃D
	7.	~ (Fa ⊃ Ga)✓	4 ∃D
	8.	Fa	7 ~ ⊃D
	9.	~ Ga	7 ~ ⊃D
	10.	~ Fa	5 ∀D
		×	

The tree is closed.

g.	1.	(∃x)Fx✓	SM
	2.	(∃y)Gy✓	SM
	3.	(∃z)(Fz & Gz)✓	SM
	4.	Fa	1 ∃D
	5.	Gb	2 ∃D
	6.	Fc & Gc✓	3 ∃D
	7.	Fc	6 &D
	8.	Gc	6 &D
		o	

The tree has a completed open branch.

i.	1.	(∀x) (∀y) (Fxy ⊃ Fyx)		SM
	2.	(∃x) (∃y) (Fxy & ~ Fyx)✓		SM
	3.	(∃y) (Fay & ~ Fya)✓		2 ∃D
	4.	Fab & ~ Fba✓		3 ∃D
	5.	Fab		4 &D
	6.	~ Fba		4 &D
	7.	(∀y) (Fay ⊃ Fya)		1 ∀D
	8.	Fab ⊃ Fba✓		7 ∀D
	9.	~ Fab	Fba	8 ⊃D
		×	×	

The tree is closed.

k.	1.	(∃x)Fx ⊃ (∀x)Fx✓		SM
	2.	~ (∀x) (Fx ⊃ (∀y)Fy)✓		SM
	3.	(∃x) ~ (Fx ⊃ (∀y)Fy)✓		2 ~ ∀D
	4.	~ (Fa ⊃ (∀y)Fy)✓		3 ∃D
	5.	Fa		4 ~ ⊃D
	6.	~ (∀y)Fy✓		4 ~ ⊃D
	7.	(∃y) ~ Fy✓		6 ~ ∀D
	8.	~ Fb		7 ∃D
	9.	~ (∃x)Fx✓	(∀x)Fx	1 ⊃D
	10.	(∀x) ~ Fx		9 ~ ∃D
	11.	~ Fa		10 ∀D
	12.	×	Fb	9 ∀D
			×	

The tree is closed.

m.	1.	(∀x) (Fx ⊃ (∃y)Gyx)		SM
	2.	~ (∀x) ~ Fx✓		SM
	3.	(∀x) (∀y) ~ Gxy		SM
	4.	(∃x) ~ ~ Fx✓		2 ~ ∀D
	5.	~ ~ Fa✓		4 ∃D
	6.	Fa		5 ~ ~ D
	7.	Fa ⊃ (∃y)Gya✓		1 ∀D
	8.	~ Fa	(∃y)Gya✓	7 ⊃D
	9.	×	Gba	8 ∃D
	10.		(∀y) ~ Gby	3 ∀D
	11.		~ Gba	10 ∀D
			×	

The tree is closed.

o.	1.	$(\exists x)Lxx \checkmark$	SM
	2.	$\sim (\exists x)(\exists y)(Lxy \;\&\; Lyx) \checkmark$	SM
	3.	$(\forall x) \sim (\exists y)(Lxy \;\&\; Lyx)$	2 ~ ∃D
	4.	Laa	1 ∃D
	5.	$\sim (\exists y)(Lay \;\&\; Lya) \checkmark$	3 ∀D
	6.	$(\forall y) \sim (Lay \;\&\; Lya)$	5 ~ ∃D
	7.	$\sim (Laa \;\&\; Laa) \checkmark$	6 ∀D
	8.	$\sim Laa$ $\sim Laa$	7 ~ &D
		× ×	

The tree is closed.

q.	1.	$(\exists x)(Fx \vee Gx) \checkmark$	SM
	2.	$(\forall x)(Fx \supset \sim Gx)$	SM
	3.	$(\forall x)(Gx \supset \sim Fx)$	SM
	4.	$\sim (\exists x)(\sim Fx \vee \sim Gx) \checkmark$	SM
	5.	$(\forall x) \sim (\sim Fx \vee \sim Gx)$	4 ~ ∃D
	6.	$Fa \vee Ga$	1 ∃D
	7.	$Fa \supset \sim Ga \checkmark$	2 ∀D
	8.	$Ga \supset \sim Fa \checkmark$	3 ∀D
	9.	$\sim (\sim Fa \vee \sim Ga) \checkmark$	5 ∀D
	10.	$\sim \sim Fa \checkmark$	9 ~ ∨D
	11.	$\sim \sim Ga \checkmark$	9 ~ ∨D
	12.	Ga	11 ~ ~ D
	13.	Fa	10 ~ ~ D
	14.	$\sim Fa$ $\sim Ga$	7 ⊃D
		× ×	

The tree is closed.

Section 9.2E

Note: In these answers, whenever a tree is open we give a complete tree. This is because the strategies we have suggested do not uniquely determine the order of decomposition, and so the first open branch to be completed on your tree may not be the first such branch completed on our tree. In accordance with strategy 5, you should stop when your tree has one completed open branch.

a. 1.	(∀x)Fx ∨ (∃y)Gy✓		SM
2.	(∃x)(Fx & Gb)✓		SM
3.	Fa & Gb✓		2 ∃D
4.	Fa		3 &D
5.	Gb		3 &D
6.	(∀x)Fx	(∃y)Gy✓	1 ∨D
7.	Fa		6 ∀D
8.	Fb		6 ∀D
9.	o	Gc	6 ∃D
		o	

The tree has two completed open branches. The set is quantificationally consistent.

c. 1.	(∀x)(Fx ⊃ Gxa)		SM
2.	(∃x)Fx✓		SM
3.	(∀y) ~ Gya		SM
4.	Fb		2 ∃D
5.	Fb ⊃ Gba✓		1 ∀D
6.	~ Fb	Gba	5 ⊃D
7.	×	~ Gba	3 ∀D
		×	

The tree is closed. The set is quantificationally inconsistent.

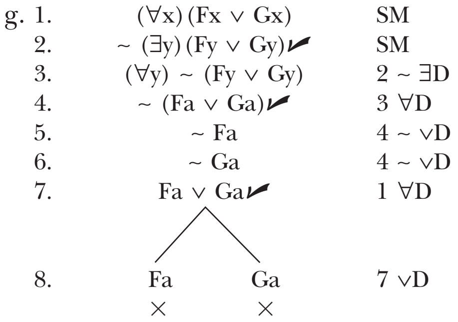
e. 1.	(∀x)(Fx ⊃ Gxa)			SM
2.	(∃x)Fx✓			SM
3.	(∀y)Gya			SM
4.	Fb			2 ∃D
5.	Fb ⊃ Gba✓			1 ∀D
6.	~ Fb	Gba		5 ⊃D
7.	×	Gaa		3 ∀D
8.		Fa ⊃ Gaa✓		1 ∀D
9.		~ Fa	Gaa	8 ⊃D
		o	o	

The tree has two completed open branches. The set is quantificationally consistent.

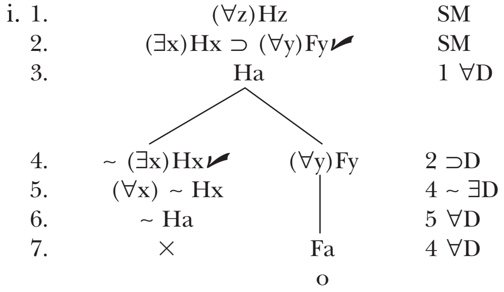
The literals ‘Fb’, ‘Gba’, ‘Gaa’, and ‘~ Fa’ on the left completed open branch will all be true on any interpretation that makes the following assignments:

UD: {1, 2}
a: 1
b: 2
Fx: x is even
Gxy: x is greater than or equal to y

The literals ‘Fb’, ‘Gba’, ‘Gaa’ on the right completed open branch will also be true on any interpretation that makes these assignments.



The tree is closed. The set is quantificationally inconsistent.

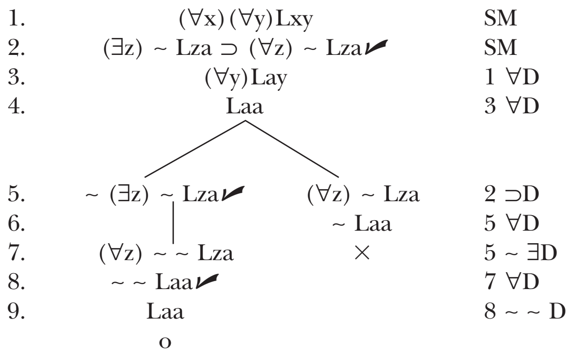


The tree has one completed open branch. The set is quantificationally consistent.

The literals ‘Ha’ and ‘Fa’ on the completed open branch will both be true on any interpretation that makes the following assignments:

UD: {1}
a: 1
Fx: x is a positive integer
Hx: x is odd

k.

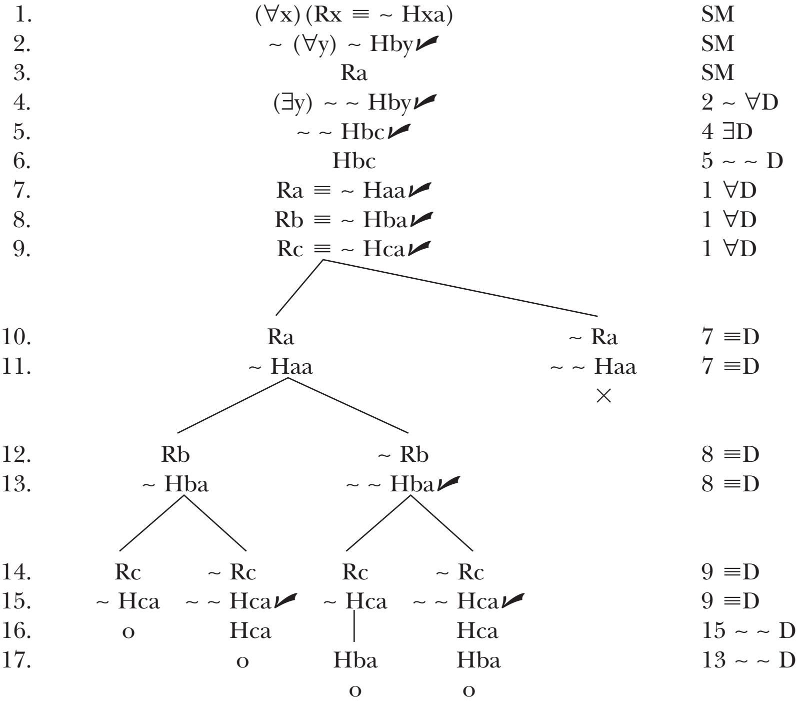


The tree has one completed open branch. The set is quantificationally consistent.

The literal 'Laa' on the completed open branch will be true on any interpretation that makes the following assignments:

UD: {1}
Lxy: x is less than or equal to y

m.



The tree has four completed open branches (the leftmost four). The set is quantificationally consistent.

The literals 'Ra', 'Rb', 'Rc', 'Hbc', '~ Haa', '~ Hba', and '~ Hca' on the leftmost completed open branch will all be true on any interpretation that makes the following assignments:

UD: {1, 2, 3}
a: 3
b: 1
c: 2
Fx: x is a positive integer
Hxy: 2 times x is equal to y

The literals 'Ra', 'Rb', 'Rc', 'Hbc', '~ Haa', '~ Hba', and '~ Hca' on the second completed open branch will all be true on any interpretation that makes the following assignments:

UD: {1, 2, 3}
a: 1
b: 2
c: 3
Rx: x is less than 3
Hxy: x + y is greater than 3

The literals 'Ra', 'Rb', 'Rc', 'Hbc', '~ Haa', '~ Hba', and '~ Hca' on the third completed open branch will all be true on any interpretation that makes the following assignments:

UD: {1, 2, 3}
a: 1
b: 3
c: 2
Rx: x is less than 3
Hxy: x + y is greater than 3

The literals 'Ra', 'Rb', 'Rc', 'Hbc', '~ Haa', '~ Hba', and '~ Hca' on the fourth completed open branch will all be true on any interpretation that makes the following assignments:

UD: {1, 2, 3}
a: 1
b: 2
c: 3
Rx: x is less than 2
Hxy: x + y is greater than 2

Section 9.3E

1.a.	1.	~ ((∃x)Fx ∨ ~ (∃x)Fx)✓	SM
	2.	~ (∃x)Fx✓	1 ~ ∨D
	3.	~ ~ (∃x)Fx✓	1 ~ ∨D
	4.	(∀x) ~ Fx	2 ~ ∃D
	5.	(∃x)Fx✓	3 ~ ~ D
	6.	Fa	5 ∃D
	7.	~ Fa	4 ∀D
		×	

The tree is closed. The sentence '(∃x)Fx ∨ ~ (∃x)Fx' is quantificationally true.

c.	1.	~ ((∀x)Fx ∨ (∀x) ~ Fx)✓	SM
	2.	~ (∀x)Fx✓	1 ~ ∨D
	3.	(∃x) ~ Fx✓	1 ~ ∨D
	4.	(∃x) ~ Fx✓	2 ~ ∀D
	5.	(∃x) ~ ~ Fx✓	3 ~ ∀D
	6.	~ Fa	4 ∃D
	7.	~ ~ Fb✓	5 ∃D
	8.	Fb	7 ~ ~ D

The tree has a completed open branch, therefore the given sentence is not quantificationally true.

e.	1.	~ ((∀x)Fx ∨ (∃x) ~ Fx)✓	SM
	2.	~ (∀x)Fx✓	1 ~ ∨D
	3.	~ (∃x) ~ Fx✓	1 ~ ∨D
	4.	(∃x) ~ Fx✓	2 ~ ∀D
	5.	(∀x) ~ ~ Fx	3 ~ ∃D
	6.	~ Fa	4 ∃D
	7.	~ ~ Fa✓	5 ∀D
	8.	Fa	7 ~ ~ D
		×	

The tree is closed. The sentence '(∀x)Fx ∨ (∃x) ~ Fx' is quantificationally true.

g.	1.	~ ((∀x)(Fx ∨ Gx) ⊃ ((∃x) ~ Fx ⊃ (∃x)Gx))✓		SM
	2.	(∀x)(Fx ∨ Gx)		1 ~ ⊃D
	3.	~ ((∃x) ~ Fx ⊃ (∃x)Gx)✓		1 ~ ⊃D
	4.	(∃x) ~ Fx✓		3 ~ ⊃D
	5.	~ (∃x)Gx✓		3 ~ ⊃D
	6.	(∀x) ~ Gx		5 ~ ∃D
	7.	~ Fa		4 ∃D
	8.	Fa ∨ Ga✓		2 ∀D
	9.	Fa	Ga	8 ∨D
	10.	×	~ Ga	6 ∀D
			×	

The tree is closed. The sentence '(∀x)(Fx ∨ Gx) ⊃ [(∃x) ~ Fx ⊃ (∃x)Gx]' is quantificationally true.

i. 1.	~ (((∀x)Fx ∨ (∀x)Gx) ⊃ (∀x)(Fx ∨ Gx))✓		SM
2.	(∀x)Fx ∨ (∀x)Gx✓		1 ~ ⊃D
3.	~ (∀x)(Fx ∨ Gx)✓		1 ~ ⊃D
4.	(∃x) ~ (Fx ∨ Gx)✓		3 ~ ∀D
5.	~ (Fa ∨ Ga)✓		4 ∃D
6.	~ Fa		5 ~ ∨D
7.	~ Ga		5 ~ ∨D
8.	(∀x)Fx	(∀x)Gx	2 ∨D
9.	Fa	Ga	8 ∀D
	×	×	

The tree is closed. The sentence '((∀x)Fx ∨ (∀x)Gx) ⊃ (∀x)(Fx ∨ Gx)' is quantificationally true.

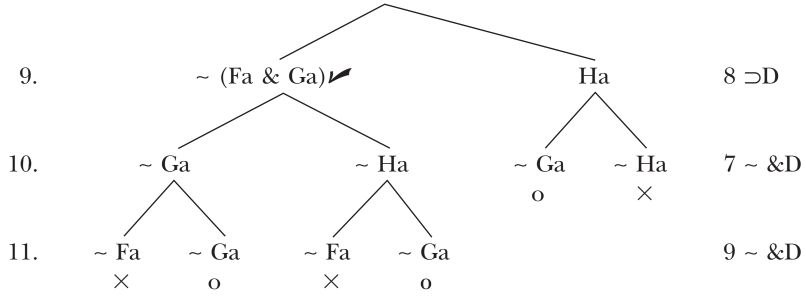
k. 1.	~ ((∃x)(Fx & Gx) ⊃ ((∃x)Fx & (∃x)Gx))✓		SM
2.	(∃x)(Fx & Gx)✓		1 ~ ⊃D
3.	~ ((∃x)Fx & (∃x)Gx)✓		1 ~ ⊃D
4.	Fa & Ga✓		2 ∃D
5.	Fa		4 &D
6.	Ga		4 &D
7.	~ (∃x)Fx✓	~ (∃x)Gx✓	3 ~ &D
8.	(∀x) ~ Fx	(∀x) ~ Gx	7 ~ ∃D
9.	~ Fa	~ Ga	8 ∀D
	×	×	

The tree is closed. The sentence '(∃x)(Fx & Gx) ⊃ ((∃x)Fx & (∃x)Gx)' is quantificationally true.

m. 1.	~ (~ (∃x)Fx ∨ (∀x) ~ Fx)✓	SM
2.	~ ~ (∃x)Fx✓	1 ~ ∨D
3.	~ (∀x) ~ Fx✓	1 ~ ∨D
4.	(∃x)Fx✓	2 ~ ~ D
5.	(∃x) ~ ~ Fx✓	3 ~ ∀D
6.	Fa	4 ∃D
7.	~ ~ Fb✓	5 ∃D
8.	Fb	7 ~ ~ D
	o	

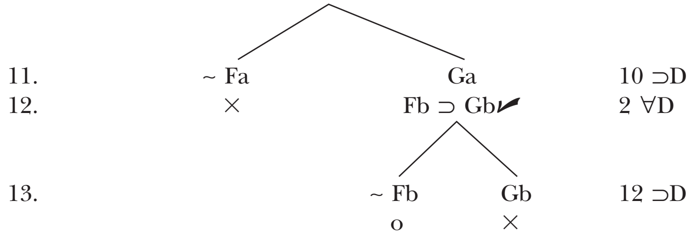
The tree has a completed open branch, therefore the given sentence is not quantificationally true.

o.	1.	~ ((∀x)((Fx & Gx) ⊃ Hx) ⊃ (∀x)(Fx ⊃ (Gx & Hx)))✓	SM
	2.	(∀x)((Fx & Gx) ⊃ Hx)	1 ~ ⊃D
	3.	~ (∀x)(Fx ⊃ (Gx & Hx))✓	1 ~ ⊃D
	4.	(∃x) ~ (Fx ⊃ (Gx & Hx))✓	3 ~ ∀D
	5.	~ (Fa ⊃ (Ga & Ha))✓	4 ∃D
	6.	Fa	5 ~ ⊃D
	7.	~ (Ga & Ha)✓	5 ~ ⊃D
	8.	(Fa & Ga) ⊃ Ha✓	2 ∀D



The tree has at least one completed open branch, therefore the given sentence is not quantificationally true.

q.	1.	~ ((∀x)(Fx ⊃ Gx) ⊃ (∀x)(Fx ⊃ (∀y)Gy))✓	SM
	2.	(∀x)(Fx ⊃ Gx)	1 ~ ⊃D
	3.	~ (∀x)(Fx ⊃ (∀y)Gy)✓	1 ~ ⊃D
	4.	(∃x) ~ (Fx ⊃ (∀y)Gy)✓	3 ~ ∀D
	5.	~ (Fa ⊃ (∀y)Gy)✓	4 ∃D
	6.	Fa	5 ~ ⊃D
	7.	~ (∀y)Gy✓	5 ~ ⊃D
	8.	(∃y) ~ Gy✓	7 ~ ∀D
	9.	~ Gb	8 ∃D
	10.	Fa ⊃ Ga✓	2 ∀D



The tree has a completed open branch, therefore the given sentence is not quantificationally true.

s. 1.	~ ((∀x)Gxx ⊃ (∀x)(∀y)Gxy)✓	SM
2.	(∀x)Gxx	1 ~ ⊃D
3.	~ (∀x)(∀y)Gxy✓	1 ~ ⊃D
4.	(∃x) ~ (∀y)Gxy✓	3 ~ ∀D
5.	~ (∀y)Gay✓	4 ∃D
6.	(∃y) ~ Gay✓	5 ~ ∀D
7.	~ Gab	6 ∃D
8.	Gaa	2 ∀D
9.	Gbb	2 ∀D
	o	

The tree has a completed open branch, therefore the given sentence is not quantificationally true.

u. 1.	~ ((∃x)(∀y)Gxy ⊃ (∀x)(∃y)Gyx)✓	SM
2.	(∃x)(∀y)Gxy✓	1 ~ ⊃D
3.	~ (∀x)(∃y)Gyx✓	1 ~ ⊃D
4.	(∃x) ~ (∃y)Gyx✓	3 ~ ∀D
5.	(∀y)Gay	2 ∃D
6.	~ (∃y)Gyb✓	4 ∃D
7.	(∀y) ~ Gyb	6 ~ ∃D
8.	Gab	5 ∀D
9.	~ Gab	7 ∀D
	×	

The tree is closed. The sentence '(∃x)(∀y)Gxy ⊃ (∀x)(∃y)Gyx' is quantificationally true.

w. 1.	~ (((∃x)Lxx ⊃ (∀y)Lyy) ⊃ (Laa ⊃ Lgg))✓		SM
2.	(∃x)Lxx ⊃ (∀y)Lyy✓		1 ~ ⊃D
3.	~ (Laa ⊃ Lgg)✓		1 ~ ⊃D
4.	Laa		3 ~ ⊃D
5.	~ Lgg		3 ~ ⊃D
6.	~ (∃x)Lxx✓	(∀y)Lyy	2 ⊃D
7.	(∀x) ~ Lxx		6 ~ ∃D
8.	~ Laa		7 ∀D
9.	×	Lgg	6 ∀D
		×	

The tree is closed. The sentence '[(∃x)Lxx ⊃ (∀y)Lyy] ⊃ (Laa ⊃ Lgg)' is quantificationally true.

2.a.	1.	(∀x)Fx & (∃x) ~ Fx✓	SM
	2.	(∀x)Fx	1 &D
	3.	(∃x) ~ Fx✓	1 &D
	4.	~ Fa	3 ∃D
	5.	Fa	2 ∀D
		×	

c.	1.	(∃x)Fx & (∃x) ~ Fx✓	SM
	2.	(∃x)Fx✓	1 &D
	3.	(∃x) ~ Fx✓	1 &D
	4.	Fa	2 ∃D
	5.	~ Fb	3 ∃D
		o	

The tree has at least one completed open branch. Therefore, the given sentence is not quantificationally false.

e.	1.	(∀x)(Fx ⊃ (∀y) ~ Fy)		SM
	2.	Fa ⊃ (∀y) ~ Fy✓		1 ∀D
	3.	~ Fa	(∀y) ~ Fy	2 ⊃D
	4.	o	~ Fa	3 ∀D
			o	

The tree has at least one completed open branch. Therefore, the given sentence is not quantificationally false.

g.	1.	(∀x)(Fx ≡ ~ Fx)		SM
	2.	Fa ≡ ~ Fa✓		1 ∀D
	3.	Fa	~ Fa	2 ≡D
	4.	~ Fa	~ ~ Fa✓	2 ≡D
	5.	×	Fa	4 ~ ~ D
			×	

The tree is closed. Therefore the sentence is quantificationally false.

i.	1.	(∃x)(∃y)(Fxy & ~ Fyx)✓	SM
	2.	(∃y)(Fay & ~ Fya)✓	1 ∃D
	3.	Fab & ~ Fba✓	2 ∃D
	4.	Fab	3 &D
	5.	~ Fba	3 &D
		o	

The tree has a completed open branch. Therefore, the given sentence is not quantificationally false.

k.	1.	(∀x)(∀y)(Fxy ⊃ ~ Fyx)		SM
	2.	(∀y)(Fay ⊃ ~ Fya)		1 ∀D
	3.	Faa ⊃ ~ Faa✓		2 ∀D
	4.	~ Faa	~ Faa	3 ⊃D
		o	o	

The tree has at least one completed open branch. Therefore, the given sentence is not quantificationally false.

m.	1.	(∃x)(∀y)Gxy & ~ (∀y)(∃x)Gxy✓	SM
	2.	(∃x)(∀y)Gxy✓	1 &D
	3.	~ (∀y)(∃x)Gxy✓	1 &D
	4.	(∃y) ~ (∃x)Gxy✓	3 ~ ∀D
	5.	(∀y)Gay	2 ∃D
	6.	~ (∃x)Gxb✓	4 ∃D
	7.	(∀x) ~ Gxb	6 ~ ∃D
	8.	Gab	5 ∀D
	9.	~ Gab	7 ∀D
		×	

The tree is closed. Therefore the sentence is quantificationally false.

3.a.	1.	~ ((∃x)Fxx ⊃ (∃x)(∃y)Fxy)✓	SM
	2.	(∃x)Fxx✓	1 ~ ⊃D
	3.	~ (∃x)(∃y)Fxy✓	1 ~ ⊃D
	4.	(∀x) ~ (∃y)Fxy	3 ~ ∃D
	5.	Faa	2 ∃D
	6.	~ (∃y)Fay✓	4 ∀D
	7.	(∀y) ~ Fay	6 ~ ∃D
	8.	~ Faa	7 ∀D
		×	

The tree for the negation of '(∃x)Fxx ⊃ (∃x)(∃y)Fxy' is closed. Therefore the latter sentence is quantificationally true.

c.	1.	~ ((∃x)(∀y)Lxy ⊃ (∃x)Lxx)✓	SM
	2.	(∃x)(∀y)Lxy✓	1 ~ ⊃D
	3.	~ (∃x)Lxx✓	1 ~ ⊃D
	4.	(∀x) ~ Lxx	3 ~ ∃D
	5.	(∀y)Lay	2 ∃D
	6.	~ Laa	4 ∀D
	7.	Laa	5 ∀D
		×	

The tree for the negation of '(∃x)(∀y)Lxy ⊃ (∃x)Lxx' is closed. Therefore the latter sentence is quantificationally true.

e.	1.	~ ((∀x)(Fx ⊃ (∃y)Gya)⊃ (Fb ⊃ (∃y)Gya))✓		SM
	2.	(∀x)(Fx ⊃ (∃y)Gya)		1 ~ ⊃D
	3.	~ (Fb ⊃ (∃y)Gya)✓		1 ~ ⊃D
	4.	Fb		3 ~ ⊃D
	5.	~ (∃y)Gya✓		3 ~ ⊃D
	6.	(∀y) ~ Gya		5 ~ ∃D
	7.	Fb ⊃ (∃y)Gya✓		2 ∀D
	8.	~ Fb	(∃y)Gya✓	7 ⊃D
	9.	×	Gca	8 ∃D
	10.		~ Gca	6 ∀D
			×	

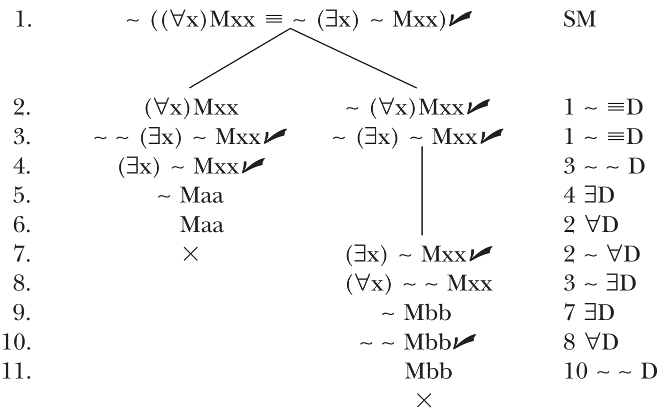
The tree for the negation of ‘(∀x)(Fx ⊃ (∃y)Gya) ⊃ (Fb ⊃ (∃y)Gya)’ is closed. Therefore the latter sentence is quantificationally true.

g.	1.	~ ((∀x)(Fx ⊃ (∀y)Gxy) ⊃ (∃x)(Fx ⊃ ~ (∀y)Gxy))✓		SM
	2.	(∀x)(Fx ⊃ (∀y)Gxy)		1 ~ ⊃D
	3.	~ (∃x)(Fx ⊃ ~ (∀y)Gxy)✓		1 ~ ⊃D
	4.	(∀x) ~ (Fx ⊃ ~ (∀y)Gxy)		3 ~ ∃D
	5.	~ (Fa ⊃ ~ (∀y)Gay)✓		4 ∀D
	6.	Fa		5 ~ ⊃D
	7.	~ ~ (∀y)Gay✓		5 ~ ⊃D
	8.	(∀y)Gay		7 ~ ~ D
	9.	Fa ⊃ (∀y)Gay✓		2 ∀D
	10.	~ Fa	(∀y)Gay	9 ⊃D
	11.	×	Gaa	10 ∀D
			o	

1.	(∀x)(Fx ⊃ (∀y)Gxy)⊃ (∃x)(Fx ⊃ ~ (∀y)Gxy)✓			SM
2.	~ (∀x)(Fx ⊃ (∀y)Gxy)✓	(∃x)(Fx ⊃ ~ (∀y)Gxy)✓		1 ⊃D
3.	(∃x) ~ (Fx ⊃ (∀y)Gxy)✓			2 ~ ∀D
4.	~ (Fa ⊃ (∀y)Gay)✓			3 ∃D
5.	Fa			4 ~ ⊃D
6.	~ (∀y)Gay✓			4 ~ ⊃D
7.	(∃y) ~ Gay✓			6 ~ ∀D
8.	~ Gab			7 ∃D
9.	o	Fa ⊃ ~ (∀y)Gay✓		2 ∃D
10.		~ Fa	~ (∀y)Gay✓	9 ⊃D
11.		o	(∃y) ~ Gay✓	10 ~ ∀D
12.			~ Gab	11 ∃D
			o	

Both the tree for the given sentence and the tree for its negation have at least one completed open branch. Therefore the given sentence is quantificationally indeterminate.

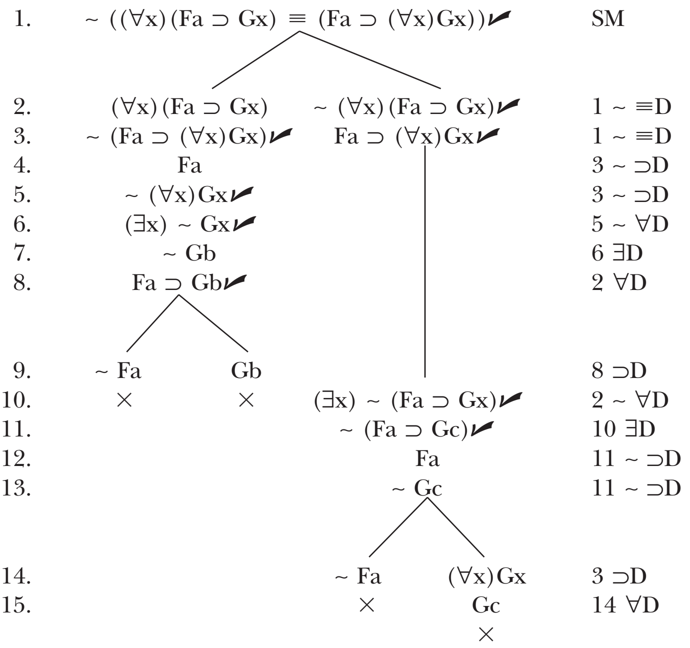
4.a.



1.	~ ((∀x)Mxx ≡ ~ (∃x) ~ Mxx)✓		SM
2.	(∀x)Mxx	~ (∀x)Mxx✓	1 ~ ≡D
3.	~ ~ (∃x) ~ Mxx✓	~ (∃x) ~ Mxx✓	1 ~ ≡D
4.	(∃x) ~ Mxx✓		3 ~ ~ D
5.	~ Maa		4 ∃D
6.	Maa		2 ∀D
7.	×	(∃x) ~ Mxx✓	2 ~ ∀D
8.		(∀x) ~ ~ Mxx	3 ~ ∃D
9.		~ Mbb	7 ∃D
10.		~ ~ Mbb✓	8 ∀D
11.		Mbb	10 ~ ~ D
		×	

The tree is closed. Therefore the sentences '(∀x)Mxx' and '~ (∃x) ~ Mxx' are quantificationally equivalent.

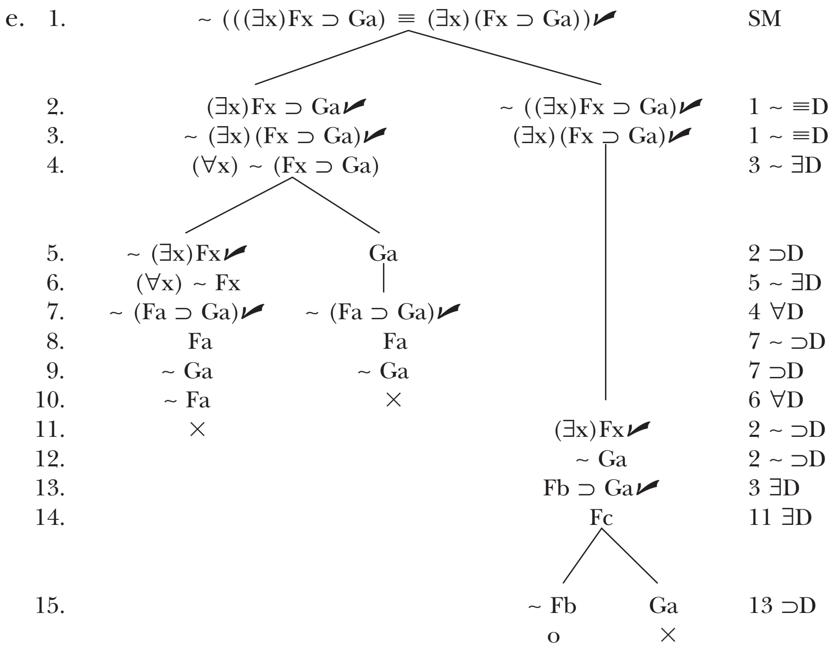
c.



1.	~ ((∀x)(Fa ⊃ Gx) ≡ (Fa ⊃ (∀x)Gx))✓				SM
2.	(∀x)(Fa ⊃ Gx)		~ (∀x)(Fa ⊃ Gx)✓		1 ~ ≡D
3.	~ (Fa ⊃ (∀x)Gx)✓		Fa ⊃ (∀x)Gx✓		1 ~ ≡D
4.	Fa				3 ~ ⊃D
5.	~ (∀x)Gx✓				3 ~ ⊃D
6.	(∃x) ~ Gx✓				5 ~ ∀D
7.	~ Gb				6 ∃D
8.	Fa ⊃ Gb✓				2 ∀D
9.	~ Fa	Gb			8 ⊃D
10.	×	×	(∃x) ~ (Fa ⊃ Gx)✓		2 ~ ∀D
11.			~ (Fa ⊃ Gc)✓		10 ∃D
12.			Fa		11 ~ ⊃D
13.			~ Gc		11 ~ ⊃D
14.			~ Fa	(∀x)Gx	3 ⊃D
15.			×	Gc	14 ∀D
				×	

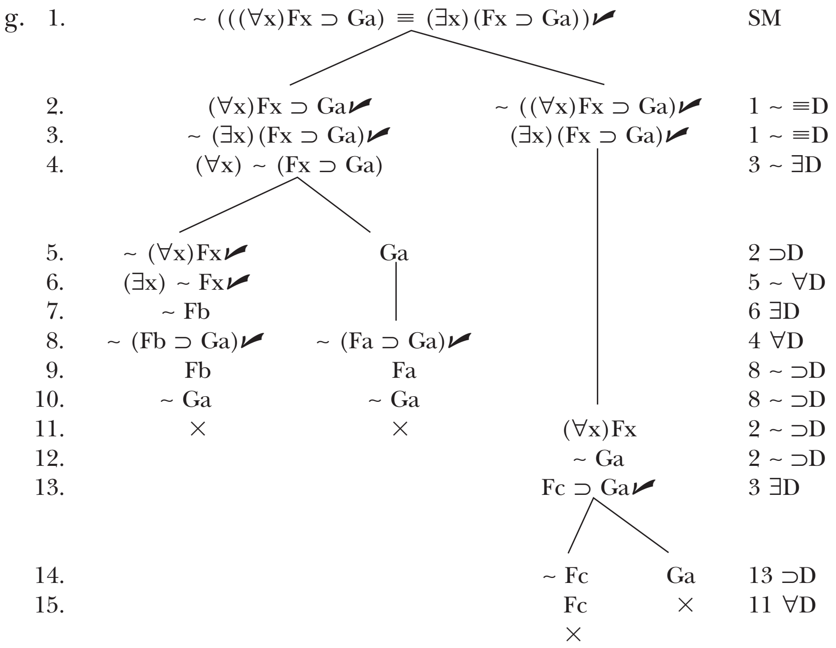
The tree is closed. Therefore the sentences '(∀x)(Fa ⊃ Gx)' and 'Fa ⊃ (∀x)Gx' are quantificationally equivalent.

e.



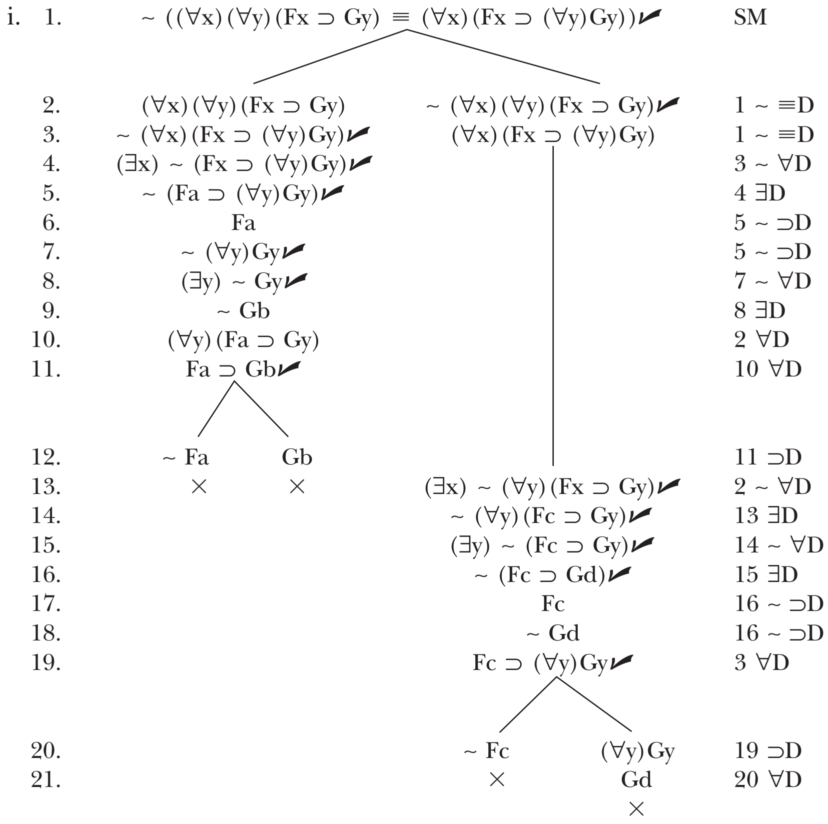
1.	~ (((∃x)Fx ⊃ Ga) ≡ (∃x)(Fx ⊃ Ga))✓				SM
2.	(∃x)Fx ⊃ Ga✓		~ ((∃x)Fx ⊃ Ga)✓		1 ~ ≡D
3.	~ (∃x)(Fx ⊃ Ga)✓		(∃x)(Fx ⊃ Ga)✓		1 ~ ≡D
4.	(∀x) ~ (Fx ⊃ Ga)				3 ~ ∃D
5.	~ (∃x)Fx✓	Ga			2 ⊃D
6.	(∀x) ~ Fx				5 ~ ∃D
7.	~ (Fa ⊃ Ga)✓	~ (Fa ⊃ Ga)✓			4 ∀D
8.	Fa	Fa			7 ~ ⊃D
9.	~ Ga	~ Ga			7 ⊃D
10.	~ Fa	×			6 ∀D
11.	×		(∃x)Fx✓		2 ~ ⊃D
12.			~ Ga		2 ~ ⊃D
13.			Fb ⊃ Ga✓		3 ∃D
14.			Fc		11 ∃D
15.			~ Fb	Ga	13 ⊃D
			o	×	

The tree has a completed open branch. Therefore the given sentences are not quantificationally equivalent.

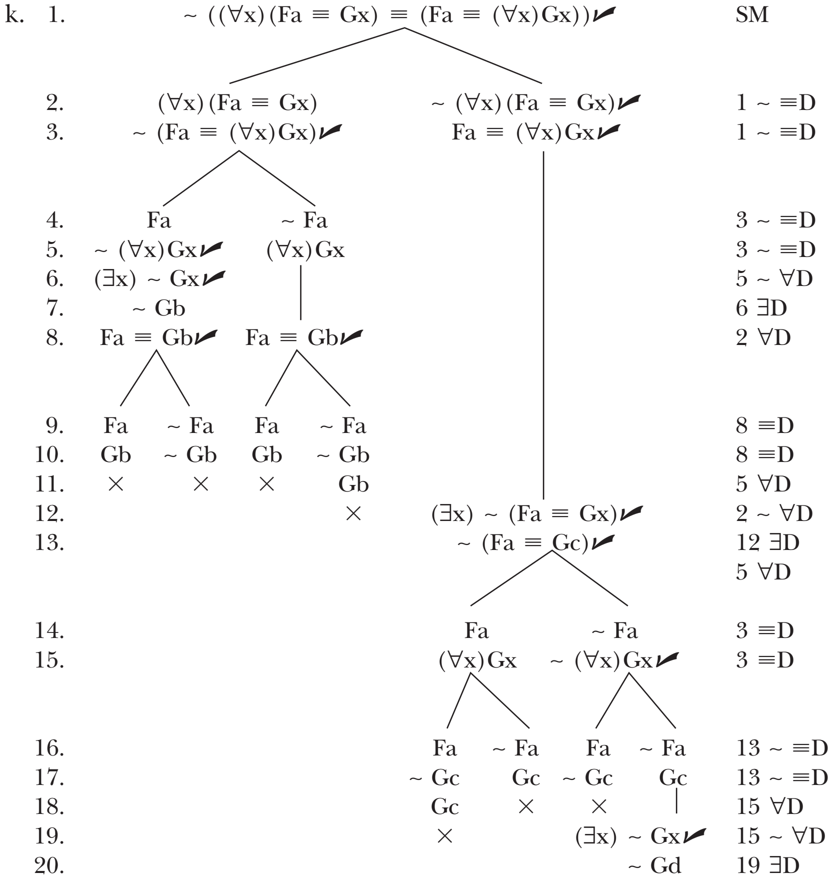


g. 1.	~ (((∀x)Fx ⊃ Ga) ≡ (∃x)(Fx ⊃ Ga))✓				SM
2.	(∀x)Fx ⊃ Ga✓		~ ((∀x)Fx ⊃ Ga)✓		1 ~ ≡D
3.	~ (∃x)(Fx ⊃ Ga)✓		(∃x)(Fx ⊃ Ga)✓		1 ~ ≡D
4.	(∀x) ~ (Fx ⊃ Ga)				3 ~ ∃D
5.	~ (∀x)Fx✓	Ga			2 ⊃D
6.	(∃x) ~ Fx✓				5 ~ ∀D
7.	~ Fb				6 ∃D
8.	~ (Fb ⊃ Ga)✓	~ (Fa ⊃ Ga)✓			4 ∀D
9.	Fb	Fa			8 ~ ⊃D
10.	~ Ga	~ Ga			8 ~ ⊃D
11.	×	×	(∀x)Fx		2 ~ ⊃D
12.			~ Ga		2 ~ ⊃D
13.			Fc ⊃ Ga✓		3 ∃D
14.			~ Fc	Ga	13 ⊃D
15.			Fc	×	11 ∀D
			×		

The tree is closed. Therefore the sentences '(∀x)Fx ⊃ Ga' and '(∃x)(Fx ⊃ Ga)' are quantificationally equivalent.



The tree is closed. Therefore the sentences ‘(∀x)(∀y)(Fx ⊃ Gy)’ and ‘(∀x)(Fx ⊃ (∀y)Gy)’ are quantificationally equivalent.



The tree has a completed open branch. Therefore the given sentences are not quantificationally equivalent.

m. 1.	~ ((∀x)(Fx ⊃ (∀y)Gy) ≡ (∀x)(∀y)(Fx ⊃ Gy))✓				SM
2.	(∀x)(Fx ⊃ (∀y)Gy)		~ (∀x)(Fx ⊃ (∀y)Gy)✓		1 ~ ≡D
3.	~ (∀x)(∀y)(Fx ⊃ Gy)✓		(∀x)(∀y)(Fx ⊃ Gy)		1 ~ ≡D
4.	(∃x) ~ (∀y)(Fx ⊃ Gy)✓				3 ~ ∀D
5.	~ (∀y)(Fa ⊃ Gy)✓				4 ∃D
6.	(∃y) ~ (Fa ⊃ Gy)✓				5 ~ ∀D
7.	~ (Fa ⊃ Gb)✓				6 ∃D
8.	Fa				7 ~ ⊃D
9.	~ Gb				7 ~ ⊃D
10.	Fa ⊃ (∀y)Gy✓				2 ∀D
11.	~ Fa	(∀y)Gy			10 ⊃D
12.	×	Gb			11 ∀D
13.		×	(∃x) ~ (Fx ⊃ (∀y)Gy)✓		2 ~ ∀D
14.			~ (Fc ⊃ (∀y)Gy)✓		13 ∃D
15.			Fc		14 ~ ⊃D
16.			~ (∀y)Gy✓		14 ~ ⊃D
17.			(∃y) ~ Gy✓		16 ~ ∀D
18.			~ Gd		17 ∃D
19.			(∀y)(Fc ⊃ Gy)		3 ∀D
20.			Fc ⊃ Gd✓		19 ∀D
21.			~ Fc	Gd	20 ⊃D
			×	×	

The tree is closed. Therefore the sentences '(∀x)(Fx ⊃ (∀y)Gy)' and '(∀x)(∀y)(Fx ⊃ Gy)' are quantificationally equivalent.

5.a. 1.	(∀x)(Fx ⊃ Gx)		SM
2.	Ga		SM
3.	~ Fa		SM
4.	Fa ⊃ Ga✓		1 ∀D
5.	~ Fa	Ga	4 ⊃D
	o	o	

The tree has at least one completed open branch. Therefore the argument is quantificationally invalid.

c.

1.	(∀x)(Kx ⊃ Lx)			SM
2.	(∀x)(Lx ⊃ Mx)			SM
3.	~ (∀x)(Kx ⊃ Mx)✓			SM
4.	(∃x) ~ (Kx ⊃ Mx)✓			3 ~ ∀D
5.	~ (Ka ⊃ Ma)✓			4 ∃D
6.	Ka			5 ~ ⊃D
7.	~ Ma			5 ~ ⊃D
8.	Ka ⊃ La✓			1 ∀D
9.	La ⊃ Ma✓			2 ∀D
10.	~ Ka ×	La		8 ⊃D
11.		~ La ×	Ma ×	9 ⊃D

The tree is closed. Therefore the argument is quantificationally valid.

e.

1.	(∀x)(Fx ⊃ Gx) ⊃ (∃x)Nx✓				SM
2.	(∀x)(Nx ⊃ Gx)				SM
3.	~ (∀x)(~ Fx ∨ Gx)✓				SM
4.	(∃x) ~ (~ Fx ∨ Gx)✓				3 ~ ∀D
5.	~ (~ Fa ∨ Ga)✓				4 ∃D
6.	~ ~ Fa✓				5 ~ ∨D
7.	~ Ga				5 ~ ∨D
8.	Fa				6 ~ ~ D
9.	Na ⊃ Ga✓				2 ∀D
10.	~ Na			Ga ×	9 ⊃D
11.	~ (∀x)(Fx ⊃ Gx)✓		(∃x)Nx✓		1 ⊃D
12.			Nb		11 ∃D
13.	(∃x) ~ (Fx ⊃ Gx)✓				11 ~ ∀D
14.	~ (Fb ⊃ Gb)✓				13 ∃D
15.	Fb				14 ~ ⊃D
16.	~ Gb				14 ~ ⊃D
17.	Nb ⊃ Gb✓		Nb ⊃ Gb✓		2 ∀D
18.	~ Nb o	Gb ×	~ Nb ×	Gb o	17 ⊃D

The tree has at least one completed open branch. Therefore the argument is quantificationally invalid.

g.	1.	(∀x)(~ Ax ⊃ Kx)		SM
	2.	(∃y) ~ Ky✓		SM
	3.	~ (∃w)(Aw ∨ ~ Lwf)✓		SM
	4.	(∀w) ~ (Aw ∨ ~ Lwf)		3 ~ ∃D
	5.	~ Ka		2 ∃D
	6.	~ Aa ⊃ Ka✓		1 ∀D
	7.	~ (Aa ∨ ~ Laf)✓		4 ∀D
	8.	~ Aa		7 ~ ∨D
	9.	~ ~ Laf✓		7 ~ ∨D
	10.	Laf		9 ~ ~ D
	11.	~ ~ Aa✓	Ka	6 ⊃D
	12.	Aa	×	11 ~ ~ D
		×		

The tree is closed. Therefore the argument is quantificationally valid.

i.	1.	(∀x)(∀y)Cxy				SM
	2.	~ ((Caa & Cab) & (Cba & Cbb))✓				SM
	3.	(∀y)Cay				1 ∀D
	4.	(∀y)Cby				1 ∀D
	5.	Caa				3 ∀D
	6.	Cab				3 ∀D
	7.	Cba				4 ∀D
	8.	Cbb				4 ∀D
	9.	~ (Caa & Cab)✓		~ (Cba & Cbb)✓		2 ~ &D
	10.	~ Caa	~ Cab	~ Cba	~ Cbb	9 ~ &D
		×	×	×	×	

The tree is closed. Therefore the argument is quantificationally valid.

k.	1.	(∀x)(Fx ⊃ Gx)		SM
	2.	~ (∃x)Fx✓		SM
	3.	~ ~ (∃x)Gx✓		SM
	4.	(∃x)Gx✓		3 ~ ~ D
	5.	Ga		4 ∃D
	6.	(∀x) ~ Fx		2 ~ ∃D
	7.	Fa ⊃ Ga✓		1 ∀D
	8.	~ Fa		6 ∀D
	9.	~ Fa	Ga	7 ⊃D
		o	o	

The tree has at least one completed open branch. Therefore the argument is quantificationally invalid.

m.

1.	(∃x)Cx ⊃ Ch✓			SM
2.	~ ((∃x)Cx ≡ Ch)✓			SM
3.	(∃x)Cx✓		~ (∃x)Cx✓	2 ~ ≡D
4.	~ Ch		Ch	2 ~ ≡D
5.			(∀x) ~ Cx	3 ~ ∃D
6.			~ Ch	5 ∀D
7.	Ca		×	3 ∃D
8.	~ (∃x)Cx✓	Ch		1 ⊃D
9.	(∀x) ~ Cx	×		8 ~ ∃D
10.	~ Ca			9 ∀D
	×			

The tree is closed. Therefore the argument is quantificationally valid.

6.a.

1.	(∀x) ~ Jx			SM
2.	(∃y)(Hby ∨ Ryy) ⊃ (∃x)Jx✓			SM
3.	~ (∀y) ~ (Hby ∨ Ryy)✓			SM
4.	(∃y) ~ ~ (Hby ∨ Ryy)✓			3 ~ ∀D
5.	~ ~ (Hba ∨ Raa)✓			4 ∃D
6.	Hba ∨ Raa✓			5 ~ ~ D
7.	~ Ja			1 ∀D
8.	~ Jb			1 ∀D
9.	~ (∃y)(Hby ∨ Ryy)✓		(∃x)Jx✓	2 ⊃D
10.			Jc	9 ∃D
11.			~ Jc	1 ∀D
12.	(∀y) ~ (Hby ∨ Ryy)		×	9 ~ ∃D
13.	~ (Hba ∨ Raa)✓			12 ∀D
14.	~ Hba			13 ~ ∨D
15.	~ Raa			13 ~ ∨D
16.	Hba	Raa		6 ∨D
	×	×		

The tree is closed. Therefore the entailment does hold.

c.	1.	(∀y)((Hy & Fy) ⊃ Gy)			SM
	2.	(∀z)Fz & ~ (∀x)Kxb✓			SM
	3.	~ (∀x)(Hx ⊃ Gx)✓			SM
	4.	(∀z)Fz			2 &D
	5.	~ (∀x)Kxb✓			2 &D
	6.	(∃x) ~ (Hx ⊃ Gx)✓			3 ~ ∀D
	7.	(∃x) ~ Kxb✓			5 ~ ∀D
	8.	~ Kab			7 ∃D
	9.	~ (Hc ⊃ Gc)✓			6 ∃D
	10.	Hc			9 ~ ⊃D
	11.	~ Gc			9 ~ ⊃D
	12.	(Hc & Fc) ⊃ Gc✓			1 ∀D
	13.	~ (Hc & Fc)✓		Gc	12 ⊃D
				×	
	14.	~ Hc	~ Fc		13 ~ &D
	15.	×	Fc		4 ∀D
			×		

The tree is closed. Therefore the entailment does hold.

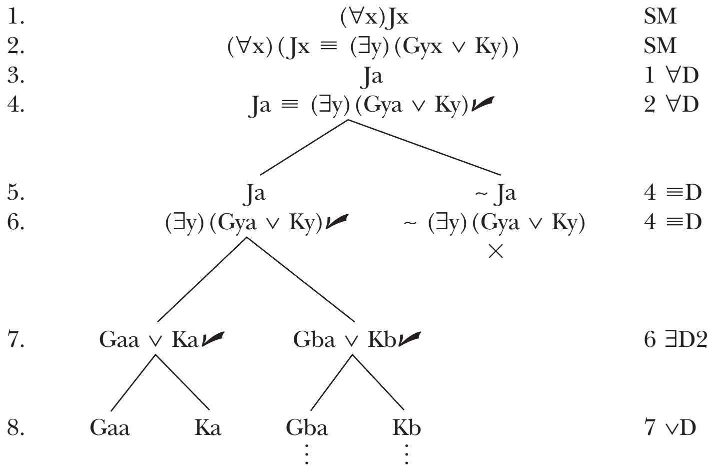
e.	1.	(∀z)(Lz ≡ Hz)		SM
	2.	(∀x) ~ (Hx ∨ ~ Bx)		SM
	3.	~ ~ Lb✓		SM
	4.	Lb		3 ~ ~ D
	5.	Lb ≡ Hb✓		1 ∀D
	6.	Lb	~ Lb	5 ≡D
	7.	Hb	~ Hb	5 ≡D
	8.	~ (Hb ∨ ~ Bb)✓	×	2 ∀D
	9.	~ Hb		8 ~ ∨D
	10.	~ ~ Bb		8 ~ ∨D
		×		

The tree is closed. Therefore the entailment does hold.

Section 9.4E

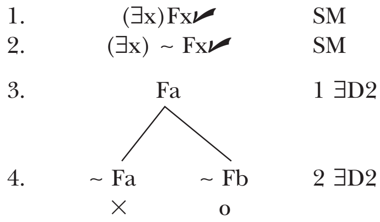
Note: Branches that are open but not completed are so indicated by a series of dots below the branch.

1.a.



The tree has at least one completed open branch. Therefore the set is quantificationally consistent.

c.



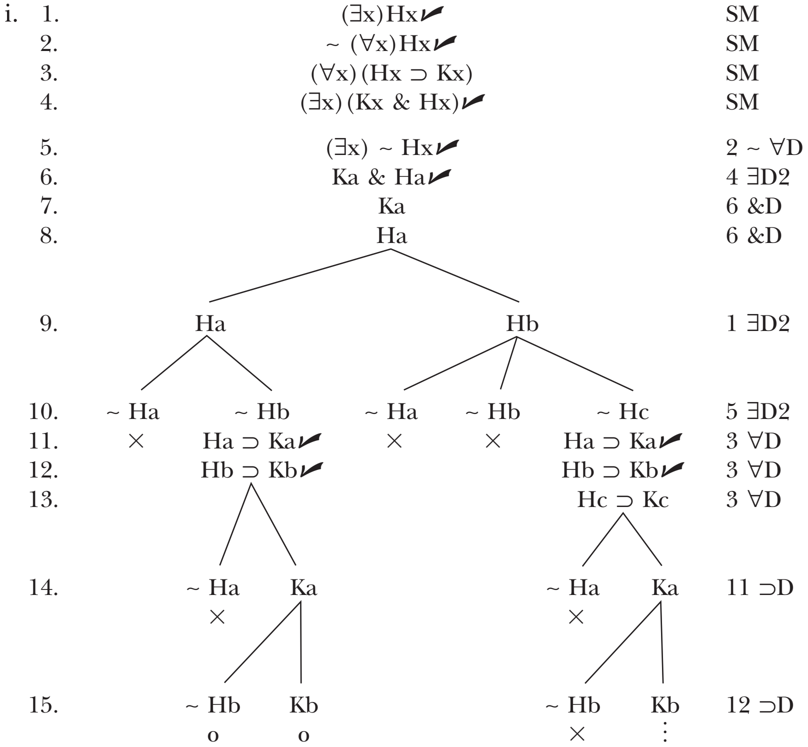
The tree has a completed open branch. Therefore the set is quantificationally consistent.

e. 1.		$(\exists x)Fx \,\&\, (\exists x) \sim Fx$ ✓		SM
2.		$(\exists x)Fx \supset (\forall x) \sim Fx$ ✓		SM
3.		$(\exists x)\ Fx$ ✓		1 &D
4.		$(\exists x) \sim Fx$ ✓		1 &D
5.		Fa		3 ∃D2
6.	$\sim Fa$	$\sim Fb$		4 ∃D2
	×			
7.		$\sim (\exists x)Fx$ ✓	$(\forall x) \sim Fx$	2 ⊃D
8.		$(\forall x) \sim Fx$		7 ~ ∃D
9.		$\sim Fa$		8 ∀D
		×	$\sim Fa$	7 ∀D
			×	

The tree is closed. Therefore the set is quantificationally inconsistent.

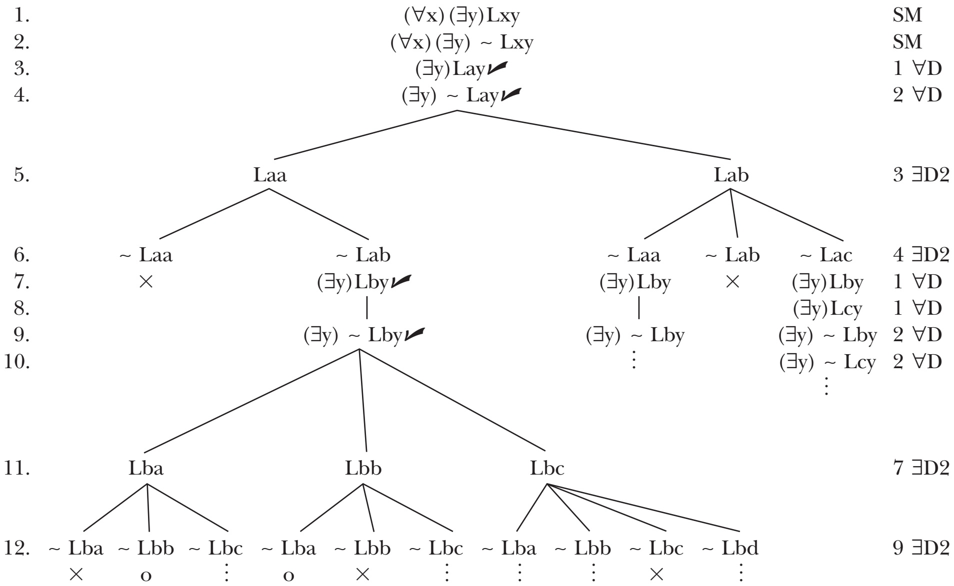
g. 1.	$(\forall x)(\exists y)Fxy$		SM
2.	$(\exists y)(\forall x) \sim Fyx$ ✓		SM
3.	$(\forall x) \sim Fax$		2 ∃D2
4.	$(\exists y)Fay$ ✓		1 ∀D
5.	$\sim Faa$		3 ∀D
6.	Faa	Fab	4 ∃D2
7.	×	$(\exists y)Fby$	1 ∀D
8.		$\sim Fab$	3 ∀D
		×	

The tree is closed. Therefore the set is quantificationally inconsistent.



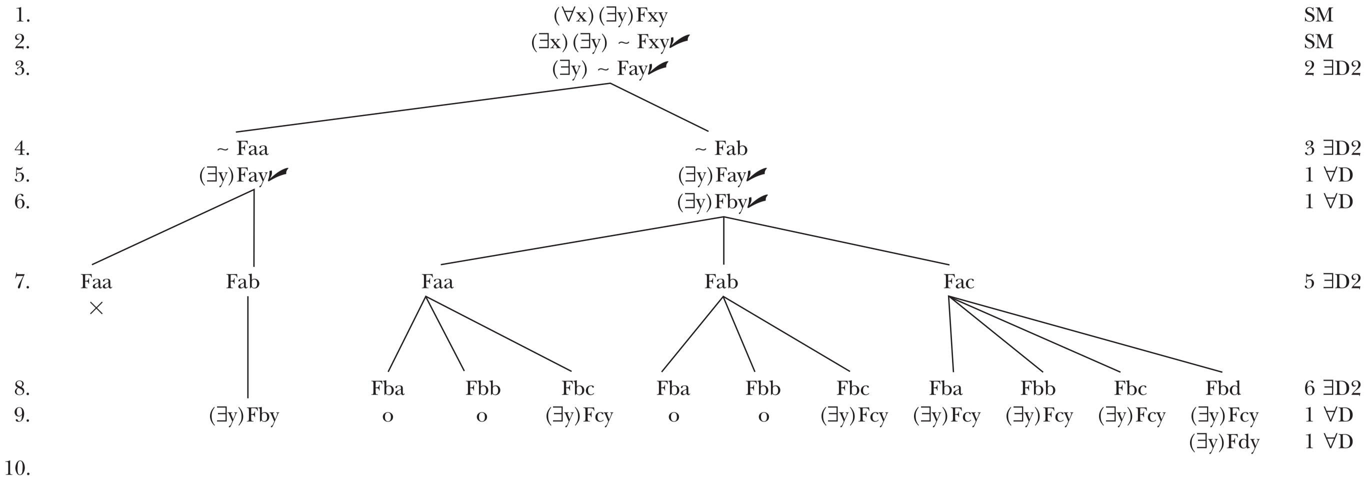
The tree has at least one completed open branch. The set is quantificationally consistent.

k.



The tree has at least one completed open branch. Therefore the set is quantificationally consistent.

m.



The tree has two completed open branch. Therefore the set is quantificationally consistent.

2.a.

1.	(∀x)(Fax ⊃ (∃y)Fya)		SM
2.	Faa ⊃ (∃y)Fya ✓		1 ∀D
3.	~ Faa	(∃y)Fya	2 ⊃D
		⋮	

1.	~ (∀x)(Fax ⊃ (∃y)Fya) ✓		SM
2.	(∃x) ~ (Fax ⊃ (∃y)Fya) ✓		1 ~ ∀D
3.	~ (Faa ⊃ (∃y)Fya) ✓	~ (Fab ⊃ (∃y)Fya) ✓	2 ∃D2
4.	Faa	Fab	3 ~ ⊃D
5.	~ (∃y)Fya ✓	~ (∃y)Fya ✓	3 ~ ⊃D
6.	(∀y) ~ Fya	(∀y) ~ Fya	5 ~ ∃D
7.	~ Faa	~ Faa	6 ∀D
8.	×	~ Fba	6 ∀D
		o	

Both the tree for the sentence and the tree for its negation have at least one completed open branch. Therefore the sentence is quantificationally indeterminate.

c.

1.	~ (∀x)(Fx ⊃ (∀y)(Hy ⊃ Fy)) ✓		SM
2.	(∃x) ~ (Fx ⊃ (∀y)(Hy ⊃ Fy)) ✓		1 ~ ∀D
3.	~ (Fa ⊃ (∀y)(Hy ⊃ Fy)) ✓		2 ∃D2
4.	Fa		3 ~ ⊃D
5.	~ (∀y)(Hy ⊃ Fy) ✓		3 ~ ⊃D
6.	(∃y) ~ (Hy ⊃ Fy) ✓		5 ~ ∀D
7.	~ (Ha ⊃ Fa) ✓	~ (Hb ⊃ Fb) ✓	6 ∃D2
8.	Ha	Hb	7 ~ ⊃D
9.	~ Fa	~ Fb	7 ~ ⊃D
	×		

1.	(∀x)(Fx ⊃ (∀y)(Hy ⊃ Fy))		SM
2.	Fa ⊃ (∀y)(Hy ⊃ Fy) ✓		1 ∀D
3.	~ Fa	(∀y)(Hy ⊃ Fy)	2 ⊃D
	o	⋮	

Both the tree for the sentence and the tree for its negation have at least one completed open branch. Therefore the sentence is quantificationally indeterminate.

e. 1.	$\sim((\exists x)(Fx \vee \sim Fx) \equiv ((\exists x)Fx \vee (\exists x)\sim Fx))\checkmark$				SM
2.	$(\exists x)(Fx \vee \sim Fx)\checkmark$		$\sim(\exists x)(Fx \vee \sim Fx)\checkmark$		$1 \sim \equiv D$
3.	$\sim((\exists x)Fx \vee (\exists x)\sim Fx)\checkmark$		$(\exists x)Fx \vee (\exists x)\sim Fx\checkmark$		$1 \sim \equiv D$
4.	$\sim(\exists x)Fx\checkmark$				$3 \sim \vee D$
5.	$\sim(\exists x)\sim Fx\checkmark$				$3 \sim \vee D$
6.	$(\forall x)\sim Fx$				$4 \sim \exists D$
7.	$(\forall x)\sim\sim Fx$				$5 \sim \exists D$
8.	$Fa \vee \sim Fa\checkmark$				$2\ \exists D2$
9.	Fa	$\sim Fa$			
10.			$(\forall x)\sim(Fx \vee \sim Fx)$		$2 \sim \exists D$
11.			$(\exists x)Fx\checkmark$	$(\exists x)\sim Fx\checkmark$	$3 \vee D$
			Fa	$\sim Fa$	$11\ \exists D2$
12.	$\sim Fa$	$\sim Fa$			$6\ \forall D$
13.	×	$\sim\sim Fa\checkmark$			$7\ \forall D$
14.		Fa			$13 \sim\sim D$
15.		×	$\sim(Fa \vee \sim Fa)\checkmark$	$\sim(Fa \vee \sim Fa)\checkmark$	$10\ \forall D$
16.			$\sim Fa$	$\sim Fa$	$15 \sim \vee D$
17.			$\sim\sim Fa$	$\sim\sim Fa\checkmark$	$15 \sim \vee D$
18.			×	Fa	$17 \sim\sim D$
				×	

The tree for the negation of the sentence is closed. Therefore the sentence is quantificationally true.

g.	1.		~ ((∀x)(Fx ⊃ ((∃y)Gyx ⊃ H)) ⊃ (∀x)(Fx ⊃ (∃y)(Gyx ⊃ H)))✓		SM
	2.		(∀x)(Fx ⊃ ((∃y)Gyx ⊃ H))		1 ~ ⊃D
	3.		~ (∀x)(Fx ⊃ ((∃y)(Gyx ⊃ H))✓		1 ~ ⊃D
	4.		(∃x) ~ (Fx ⊃ (∃y)(Gyx ⊃ H))✓		3 ~ ∀D
	5.		~ (Fa ⊃ (∃y)(Gya ⊃ H))✓		4 ∃D2
	6.		Fa		5 ~ ⊃D
	7.		~ (∃y)(Gya ⊃ H)✓		5 ~ ⊃D
	8.		(∀y) ~ (Gya ⊃ H)		7 ~ ∃D
	9.		~ (Gaa ⊃ H)✓		8 ∀D
	10.		Fa ⊃ ((∃y)Gya ⊃ H)✓		2 ∀D
	11.		Gaa		9 ~ ⊃D
	12.		~ H		9 ~ ⊃D
	13.	~ Fa	(∃y)Gya ⊃ H✓		10 ⊃D
		×			
	14.		~ (∃y)Gya✓	H	13 ⊃D
	15.		(∀y) ~ Gya	×	14 ~ ∃D
	16.		~ Gaa		15 ∀D
			×		

The tree for the negation of the sentence is closed. Therefore the sentence is quantificationally true.

3.a.	1.					Fa					SM
	2.					(∀x)(Fx ⊃ Cx)					SM
	3.					~ (∀x)(Fx & Cx)✓					SM
	4.					(∃x) ~ (Fx & Cx)✓					2 ~ ∀D
	5.		~ (Fa & Ca)✓					~ (Fb & Cb)✓			4 ∃D2
	6.	~ Fa	~ Ca		~ Fb			~ Cb			5 ~ &D
	7.	×	Fa ⊃ Ca✓		Fa ⊃ Ca✓			Fa ⊃ Ca✓			2 ∀D
	8.				Fb ⊃ Cb✓			Fb ⊃ Cb✓			2 ∀D
	9.		~ Fa	Ca	~ Fa	Ca		~ Fa	Ca		7 ⊃D
			×	×	×			×			
	10.					~ Fb	Cb		~ Fb	Cb	8 ⊃D
						o	o			×	

The tree for the premises and the negation of the conclusion has at least one completed open branch. Therefore the argument is quantificationally invalid.

c. 1. Fa SM
2. (∀x)(Fx ⊃ Cx) SM
3. ~ (∃x)(Fx & Cx)✓ SM
4. (∀x) ~ (Fx & Cx) 3 ~ ∃D
5. Fa ⊃ Ca✓ 2 ∀D
6. ~ (Fa & Ca)✓ 4 ∀D

7. ~ Fa Ca 5 ⊃D
×

8. ~ Fa ~ Ca 6 ~ &D
× ×

The tree for the premises and the negation of the conclusion is closed. Therefore the argument is quantificationally valid.

e. 1. (∀x)(∀y)(∀z)((Lxy & Lyz) ⊃ Lxz) SM
2. (∀x)(∀y)(Lxy ⊃ Lyx) SM
3. ~ (∀x)Lxx✓ SM
4. (∃x) ~ Lxx✓ 3 ~ ∀D
5. ~ Laa 4 ∃D2
6. (∀y)(∀z)((Lay & Lyz)⊃ Laz) 1 ∀D
7. (∀y)(Lay ⊃ Lya) 2 ∀D
8. (∀z)((Laa & Laz) ⊃ Laz) 6 ∀D
9. Laa ⊃ Laa✓ 7 ∀D
10. (Laa & Laa) ⊃ Laa✓ 8 ∀D

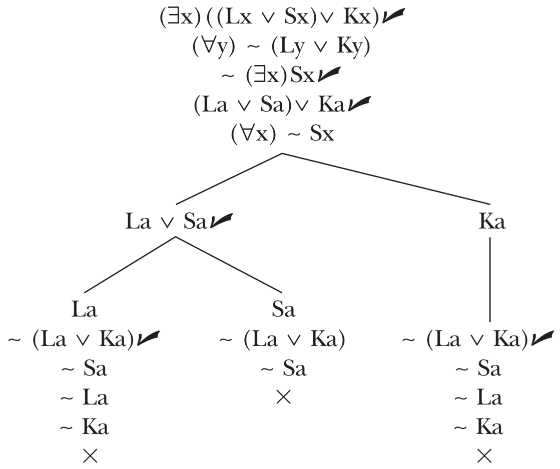
11. ~ (Laa & Laa)✓ Laa 10 ⊃D
×

12. ~ Laa Laa 9 ⊃D
×

13. ~ Laa ~ Laa 11 ~ &D
o o

The tree for the premises and the negation of the conclusion has at least one completed open branch. Therefore the argument is quantificationally invalid.

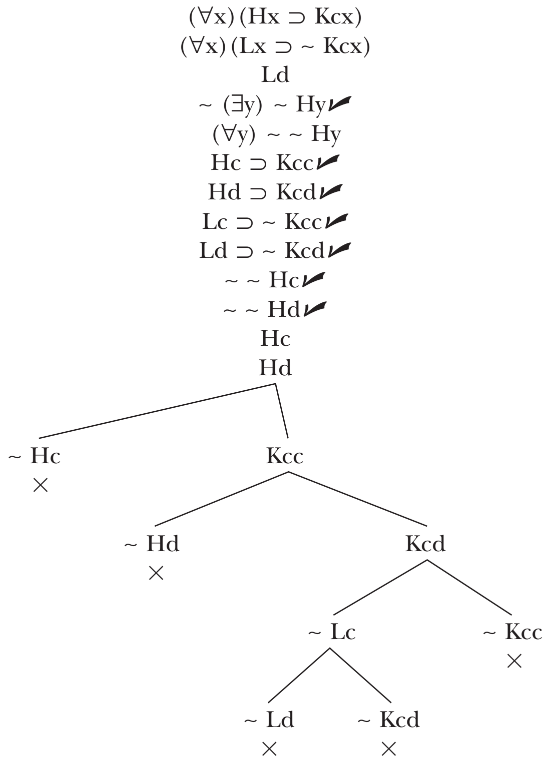
g.



1.		$(\exists x)((Lx \vee Sx) \vee Kx)$ ✓		SM
2.		$(\forall y) \sim (Ly \vee Ky)$		SM
3.		$\sim (\exists x)Sx$ ✓		SM
4.		$(La \vee Sa) \vee Ka$ ✓		1 ∃D2
5.		$(\forall x) \sim Sx$		3 ~ ∃D
6.	$La \vee Sa$ ✓		Ka	4 ∨D
7.	La	Sa		6 ∨D
8.	$\sim (La \vee Ka)$ ✓	$\sim (La \vee Ka)$	$\sim (La \vee Ka)$ ✓	2 ∀D
9.	$\sim Sa$	$\sim Sa$	$\sim Sa$	5 ∀D
10.	$\sim La$	×	$\sim La$	8 ~ ∨D
11.	$\sim Ka$		$\sim Ka$	8 ~ ∨D
	×		×	

The tree for the premises and the negation of the conclusion is closed. Therefore the argument is quantificationally valid.

i.



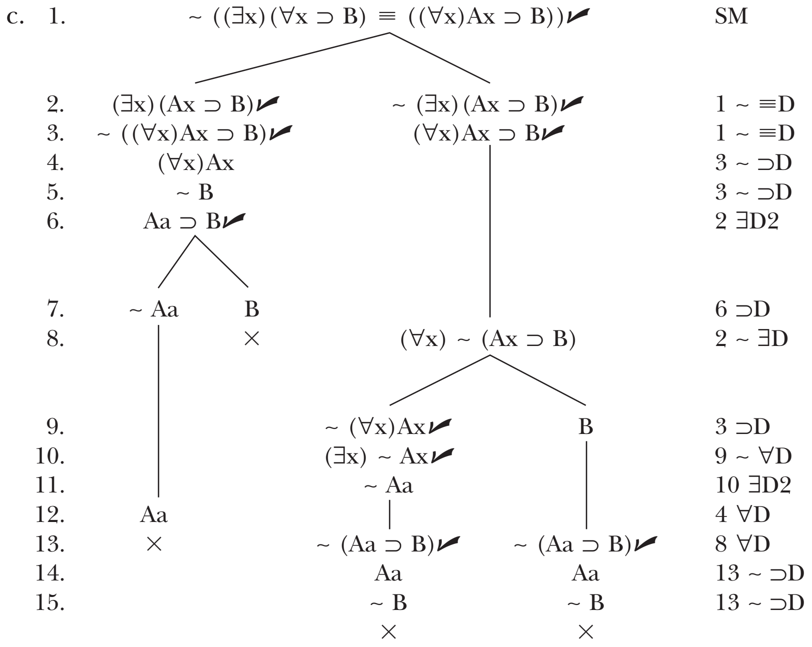
1.			$(\forall x)(Hx \supset Kcx)$			SM
2.			$(\forall x)(Lx \supset \sim Kcx)$			SM
3.			Ld			SM
4.			$\sim (\exists y) \sim Hy$ ✓			SM
5.			$(\forall y) \sim \sim Hy$			4 ~ ∃D
6.			$Hc \supset Kcc$ ✓			1 ∀D
7.			$Hd \supset Kcd$ ✓			1 ∀D
8.			$Lc \supset \sim Kcc$ ✓			2 ∀D
9.			$Ld \supset \sim Kcd$ ✓			2 ∀D
10.			$\sim \sim Hc$ ✓			5 ∀D
11.			$\sim \sim Hd$ ✓			5 ∀D
12.			Hc			10 ~ ~ D
13.			Hd			11 ~ ~ D
14.	$\sim Hc$		Kcc			6 ⊃D
	×					
15.		$\sim Hd$		Kcd		7 ⊃D
		×				
16.			$\sim Lc$		$\sim Kcc$	8 ⊃D
					×	
17.			$\sim Ld$	$\sim Kcd$		9 ⊃D
			×	×		

The tree for the premises and the negation of the conclusion is closed. Therefore the argument is quantificationally valid.

6.a.

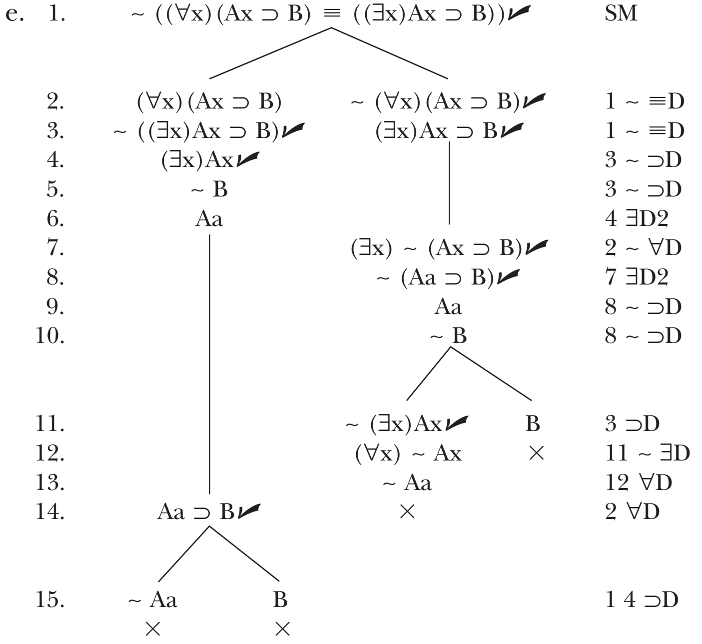
1.	~ ((∀x)(∀y) ~ Sxy ≡ ~ (∃x)(∃y)Sxy)✓				SM
2.	(∀x)(∀y) ~ Sxy		~ (∀x)(∀y) ~ Sxy✓		1 ~ ≡D
3.	~ ~ (∃x)(∃y)Sxy✓		~ (∃x)(∃y)Sxy✓		1 ~ ≡D
4.	(∃x)(∃y)Sxy✓				3 ~ ~ D
5.	(∃y)Say✓				4 ∃D2
6.	Saa	Sab			5 ∃D2
7.			(∃x) ~ (∀y) ~ Sxy✓		2 ~ ∀D
8.			(∀x) ~ (∃y)Sxy		3 ~ ∃D
9.			~ (∀y) ~ Say✓		7 ∃D2
10.			(∃y) ~ ~ Say✓		9 ~ ∀D
11.			~ ~ Saa✓	~ ~ Sab✓	10 ∃D2
12.			Saa	Sab	11 ~ ~ D
13.			~ (∃y)Say✓	~ (∃y)Say✓	8 ∀D
14.				~ (∃y)Sby✓	8 ∀D
15.	(∀y) ~ Say	(∀y) ~ Say			2 ∀D
16.		(∀y) ~ Sby			2 ∀D
17.	~ Saa	~ Sab			15 ∀D
18.	×	×	(∀y) ~ Say	(∀y) ~ Say	13 ~ ∃D
19.				(∀y) ~ Sby	14 ~ ∃D
20.			~ Saa	~ Sab	18 ∀D
			×	×	

The tree for the negation of the corresponding biconditional is closed. Therefore the sentences are equivalent.



c.	1.	$\sim ((\exists x)(\forall x \supset B) \equiv ((\forall x)Ax \supset B))$✓				SM
	2.	$(\exists x)(Ax \supset B)$✓		$\sim (\exists x)(Ax \supset B)$✓		$1 \sim \equiv D$
	3.	$\sim ((\forall x)Ax \supset B)$✓		$(\forall x)Ax \supset B$✓		$1 \sim \equiv D$
	4.	$(\forall x)Ax$				$3 \sim \supset D$
	5.	$\sim B$				$3 \sim \supset D$
	6.	$Aa \supset B$✓				2 $\exists D2$
	7.	$\sim Aa$	B			6 $\supset D$
	8.		×	$(\forall x) \sim (Ax \supset B)$		$2 \sim \exists D$
	9.			$\sim (\forall x)Ax$✓	B	3 $\supset D$
	10.			$(\exists x) \sim Ax$✓		$9 \sim \forall D$
	11.			$\sim Aa$		10 $\exists D2$
	12.	Aa				4 $\forall D$
	13.	×		$\sim (Aa \supset B)$✓	$\sim (Aa \supset B)$✓	8 $\forall D$
	14.			Aa	Aa	$13 \sim \supset D$
	15.			$\sim B$	$\sim B$	$13 \sim \supset D$
				×	×	

The tree for the negation of the corresponding biconditional is closed. Therefore the sentences are equivalent.



e.	1.	$\sim ((\forall x)(Ax \supset B) \equiv ((\exists x)Ax \supset B))$✓				SM
	2.	$(\forall x)(Ax \supset B)$		$\sim (\forall x)(Ax \supset B)$✓		$1 \sim \equiv D$
	3.	$\sim ((\exists x)Ax \supset B)$✓		$(\exists x)Ax \supset B$✓		$1 \sim \equiv D$
	4.	$(\exists x)Ax$✓				$3 \sim \supset D$
	5.	$\sim B$				$3 \sim \supset D$
	6.	Aa				4 $\exists D2$
	7.			$(\exists x) \sim (Ax \supset B)$✓		$2 \sim \forall D$
	8.			$\sim (Aa \supset B)$✓		7 $\exists D2$
	9.			Aa		$8 \sim \supset D$
	10.			$\sim B$		$8 \sim \supset D$
	11.			$\sim (\exists x)Ax$✓	B	3 $\supset D$
	12.			$(\forall x) \sim Ax$	×	$11 \sim \exists D$
	13.			$\sim Aa$		12 $\forall D$
	14.	$Aa \supset B$✓		×		2 $\forall D$
	15.	$\sim Aa$	B			1 4 $\supset D$
		×	×			

The tree for the negation of the corresponding biconditional is closed. Therefore the sentences are equivalent.

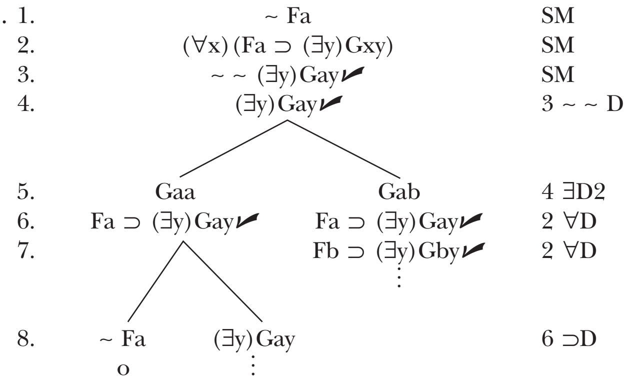
g. 1.	~ ((∃x)(∃y)Hxy ≡ (∃y)(∃x)Hxy)✓				SM
2.	(∃x)(∃y)Hxy✓		~ (∃x)(∃y)Hxy✓		1 ~ ≡D
3.	~ (∃y)(∃x)Hxy✓		(∃y)(∃x)Hxy✓		1 ~ ≡D
4.	(∃y)Hay✓				2 ∃D2
5.	(∀y) ~ (∃x)Hxy				3 ~ ∃D
6.			(∃x)Hxa✓		3 ∃D2
7.	Haa	Hab			4 ∃D2
8.			(∀x) ~ (∃y)Hxy		2 ~ ∃D
9.			Haa	Hba	6 ∃D2
10.	~ (∃x)Hxa✓	~ (∃x)Hxa✓			5 ∀D
11.		~ (∃x)Hxb✓			5 ∀D
12.			~ (∃y)Hay✓	~ (∃y)Hay✓	8 ∀D
13.				~ (∃y)Hby✓	8 ∀D
14.	(∀x) ~ Hxa	(∀x) ~ Hxa			10 ~ ∃D
15.		(∀x) ~ Hxb			11 ~ ∃D
16.			(∀y) ~ Hay	(∀y) ~ Hay	12 ~ ∃D
17.				(∀y) ~ Hby	13 ~ ∃D
18.	~ Haa	~ Haa			14 ∀D
19.	×	~ Hab			15 ∀D
20.		×	~ Haa		16 ∀D
21.			×	~ Hba	17 ∀D
				×	

The tree for the negation of the corresponding biconditional is closed. Therefore the sentences are equivalent.

5.a. 1.	(∀x)(Fax ⊃ Fxa)			SM
2.	~ (Fab ∨ Fba)✓			SM
3.	~ Fab			2 ~ ∨D
4.	~ Fba			2 ~ ∨D
5.	Faa ⊃ Faa✓			1 ∀D
6.	Fab ⊃ Fba✓			1 ∀D
7.	~ Fab		Fba ×	6 ⊃D
8.	~ Faa o	Faa o		5 ⊃D

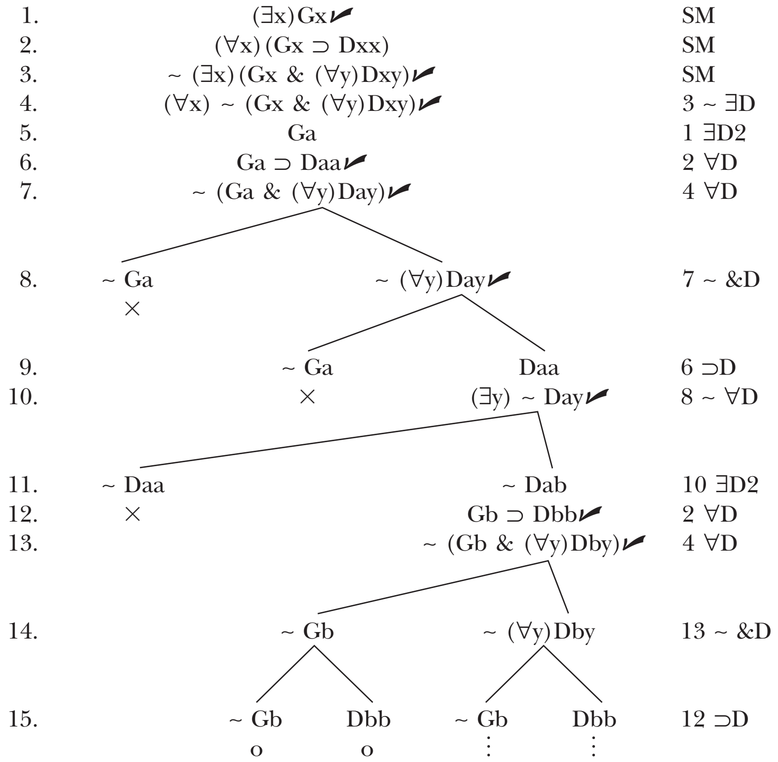
The tree has at least one completed open branch. Therefore the given set does not quantificationally entail the given sentence.

c.



The tree has at least one completed open branch. Therefore the given set does not quantificationally entail the given sentence.

e.



The tree has at least one completed open branch. Therefore the given set does not quantificationally entail the given sentence.

7. If a tree is closed, then on each branch of that tree there is some atomic sentence **P** and its negation, ~ **P**. One of these sentences occurs subsequent to the other on the branch in question. Let **Q** be the latter of the two sentences and let **n** be the number of the line on which **Q** occurs. Then **n** is either the last line of the branch or the second to the last line of the branch. The reason is that once both an atomic sentence and its negation have been added to a branch, that branch is closed and no further sentences can be added to the branch after the current decomposition has been completed. (Some decomposition rules do add two sentences to each branch passing through the sentence being decomposed.) Hence such a branch is finite—for no infinite branch can have a last member.

9. No. For example, consider the sentence '(∃x)(Fx & ~ Fb)' and its substitution instance 'Fb & ~ Fb'. Clearly, every tree for the unit set of the latter sentence closes, but the systematic tree for the unit set of '(∃x)(Fx & ~ Fb)' does not close. Rather it has a completed open branch:

1.	(∃x)(Fx & ~ Fb)✓		SM
2.	Fa & ~ Fb✓	Fb & ~ Fb✓	1 ∃D2
3.	Fa	Fb	2 &D
4.	~ Fb	~ Fb	2 &D
		×	

11. Since it is already specified that stage 1 is done before stage 2 and stage 2 before stage 3, and stage 3 before stage 4, we would have to specify the order in which work within each stage is to be done, and what constants are to be used in what order.

Section 9.5E

1.a.

1.	(∀x)Fxx	SM
2.	(∃x)(∃y) ~ Fxy✓	SM
3.	(∀x)x = a	SM
4.	(∃y) ~ Fby✓	2 ∃D
5.	~ Fbc	4 ∃D
6.	Faa	1 ∀D
7.	c = a	3 ∀D
8.	Fac	6, 7 =D
9.	b = a	3 ∀D
10.	Fbc	8, 9 =D
	×	

The tree is closed. Therefore the set is quantificationally inconsistent.

c. 1.	(∀x)(x = a ⊃ Gxb)		SM
2.	~ (∃x)Gxx✓		SM
3.	a = b		SM
4.	(∀x) ~ Gxx		2 ~ ∃D
5.	a = a ⊃ Gab✓		1 ∀D
6.	~ a = a	Gab	5 ⊃D
7.	×	~ Gaa	4 ∀D
8.		Gaa	3, 6 =D
		×	

The tree is closed. Therefore the set is quantificationally inconsistent.

e. 1.	(∀x)((Fx & ~ Gx) ⊃ ~ x = a)			SM
2.	Fa & ~ Ga✓			SM
3.	Fa			2 &D
4.	~ Ga			2 &D
5.	(Fa & ~ Ga) ⊃ ~ a = a✓			1 ∀D
6.	~ (Fa & ~ Ga)✓		~ a = a	5 ⊃D
			×	
7.	~ Fa	~ ~ Ga✓		6 ~ &D
8.	×	Ga		7 ~ ~ D
		×		

The tree is closed. Therefore the set is quantificationally inconsistent.

g. 1.	(∀x)(x = a ⊃ Gxf(b))		SM
2.	~ (∃x)Gxf(x)✓		SM
3.	f(a)= f(b)		SM
4.	(∀x) ~ Gxf(x)		2 ~ ∃D
5.	a = a ⊃ Gaf(b)✓		1 ∀D
6.	~ a = a	Gaf(b)	5 ⊃D
7.	×	~ Gaf(a)	4 ∀D
8.		Gaf(a)	3, 6 =D
		×	

The tree is closed. Therefore the set is quantificationally inconsistent.

i. 1.	(∃x) ~ x = *g*(x) ✓	SM
2.	(∀x)(∀y)x = *g*(y)	SM
3.	~ a = *g*(a)	1 ∃D
4.	(∀y)a = *g*(y)	2 ∀D
5.	a = *g*(a)	4 ∀D
	×	

The tree is closed. Therefore the set is quantificationally inconsistent.

k. 1.	(∀x)[Hx ⊃ (∀y)Txy]		SM
2.	(∃x)H*f*(x) ✓		SM
3.	~ (∃x)Txx ✓		SM
4.	H*f*(a)		2 ∃D
5.	(∀x) ~ Txx		3 ~ ∃D
6.	H*f*(a) ⊃ (∀y)T*f*(a)y ✓		1 ∀D
			6 ⊃D
7.	~ H*f*(a)	(∀y)T*f*(a)y)	
	×	T*f*(a)*f*(a)	7 ∀D
		~ T*f*(a)*f*(a)	5 ∀D
		×	

The tree is closed. Therefore the set is quantificationally inconsistent.

m. 1.	(∃x)Fx ⊃ (∃x)(∃y)*f*(y)= x ✓		SM
2.	(∃x)Fx ✓		SM
3.	Fa		2 ∃D
4.	~(∃x)Fx ✓	(∃x)(∃y)*f*(y)= x ✓	1 ⊃D
5.	(∀x)~Fx		4 ~ ∃
6.	~Fa		5 ∀D
7.	×	(∃y)*f*(y)= b ✓	4 ∃D
8.		*f*(c)= b	7 ∃D
		o	

The tree has a completed open branch. Therefore the set is quantificationally consistent.

The literals ‘Fa’, and ‘*f*(c) = b’ on the completed open branch will be be true on any interpretation that makes the following assignments:

UD: {2, 4, 6}
a: 6
b: 4
c: 2
f(x): x^2
Fx: x is even

2.a.

1.	$\sim (a = b \equiv b = a)$✓		SM
2.	$a = b$	$\sim a = b$	$1 \sim \equiv D$
3.	$\sim b = a$	$b = a$	$1 \sim \equiv D$
4.	$\sim a = a$	$\sim b = b$	$2, 3 = D$
	×	×	

The tree is closed. Therefore ‘$a = b \equiv b = a$’ is quantificationally true.

c.

1.	$\sim ((Gab \,\&\, \sim Gba) \supset \sim a = b)$✓	SM
2.	$Gab \,\&\, \sim Gba$✓	$1 \sim \supset D$
3.	$\sim \sim a = b$✓	$1 \sim \supset D$
4.	Gab	2 &D
5.	$\sim Gba$	2 &D
6.	$a = b$	$3 \sim \sim D$
7.	Gaa	$4, 6 = D$
8.	$\sim Gaa$	$5, 6 = D$
	×	

The tree is closed. Therefore the sentence ‘$(Gab \,\&\, \sim Gba) \supset \sim a = b$’ is quantificationally true.

e.

1.	$\sim (Fa \equiv (\exists x)(Fx \,\&\, x = a))$✓			SM
2.	Fa		$\sim Fa$	$1 \sim \equiv D$
3.	$\sim (\exists x)(Fx \,\&\, x = a)$✓		$(\exists x)(Fx \,\&\, x = a)$✓	$1 \sim \equiv D$
4.	$(\forall x) \sim (Fx \,\&\, x = a)$			$3 \sim \exists D$
5.	$\sim (Fa \,\&\, a = a)$✓			$4 \forall D$
6.	$\sim Fa$	$\sim a = a$		$5 \sim \&D$
7.	×	×	$Fb \,\&\, b = a$✓	$3 \exists D$
8.			Fb	7 &D
9.			$b = a$	7 &D
10.			$\sim Fb$	$2, 9 = D$
			×	

The tree is closed. Therefore the sentence ‘$Fa \equiv (\exists x)(Fx \,\&\, x = a)$’ is quantificationally true.

g.	1.	~ ((∀x)x = a ⊃ ((∃x)Fx ⊃ (∀x)Fx))✓	SM
	2.	(∀x)x = a	1 ~ ⊃D
	3.	~ ((∃x)Fx ⊃ (∀x)Fx)✓	1 ~ ⊃D
	4.	(∃x)Fx✓	3 ~ ⊃D
	5.	~ (∀x)Fx✓	3 ~ ⊃D
	6.	(∃x) ~ Fx✓	5 ~ ∀D
	7.	Fb	4 ∃D
	8.	~ Fc	6 ∃D
	9.	c = a	2 ∀D
	10.	b = a	2 ∀D
	11.	c = b	9, 10 =D
	12.	Fc	7, 11 =D
		×	

The tree is closed. Therefore the sentence '(∀x)x = a ⊃ ((∃x)Fx ⊃ (∀x)Fx)' is quantificationally true.

i.	1.	(∀x)(∀y) ~ x = y	SM
	2.	(∀y) ~ a = y	1 ∀D
	3.	~ a = a	2 ∀D
		×	

The tree is closed. Therefore the sentence '(∀x)(∀y) ~ x = y' is quantificationally false.

k.	1.	(∃x)(∃y) ~ x = y✓	SM
	2.	(∃y) ~ a = y✓	1 ∃D
	3.	~ a = b	2 ∃D

1.	~ (∃x)(∃y) ~ x = y✓	SM
2.	(∀x) ~ (∃y) ~ x = y	1 ~ ∃D
3.	~ (∃y) ~ a = y✓	2 ∀D
4.	(∀y) ~ ~ a = y	3 ~ ∃D
5.	~ ~ a = a✓	4 ∀D
6.	a = a	5 ~ ~ D
	o	

Both the tree for the given sentence and the tree for its negation have at least one completed open branch. Therefore the given sentence is quantificationally indeterminate.

m. 1.	~ (∀x)(∀y)((Fx ≡ Fy) ⊃ x = y)✓		SM
2.	(∃x) ~ (∀y)((Fx ≡ Fy) ⊃ x = y)✓		1 ~ ∀D
3.	~ (∀y)((Fa ≡ Fy) ⊃ a = y)✓		2 ∃D
4.	(∃y) ~ ((Fa ≡ Fy) ⊃ a = y)✓		3 ~ ∀D
5.	~ ((Fa ≡ Fb) ⊃ a = b)✓		4 ∃D
6.	Fa ≡ Fb✓		5 ~ ⊃D
7.	~ a = b		5 ~ ⊃D
8.	Fa	~ Fa	6 ≡D
9.	Fb	~ Fb	6 ≡D

1.	(∀x)(∀y)((Fx ≡ Fy) ⊃ x = y)			SM
2.	(∀y)((Fa ≡ Fy) ⊃ a = y)			1 ∀D
3.	(Fa ≡ Fa) ⊃ a = a✓			2 ∀D
4.	~ (Fa ≡ Fa)✓		a = a	3 ⊃D
			o	
5.	Fa	~ Fa		4 ~ ≡D
6.	~ Fa	Fa		4 ~ ≡D
	×	×		

Both the tree for the given sentence and the tree for its negation have at least one completed open branch. Therefore the given sentence is quantificationally indeterminate.

o. 1.	~ (((∃x)Gax & ~ (∃x)Gxa) ⊃ (∀x)(Gxa ⊃ ~ x = a))✓	SM
2.	(∃x)Gax & ~ (∃x)Gxa✓	1 ~ ⊃D
3.	~ (∀x)(Gxa ⊃ ~ x = a)✓	1 ~ ⊃D
4.	(∃x)Gax✓	2 &D
5.	~ (∃x)Gxa✓	2 &D
6.	(∀x) ~ Gxa	5 ~ ∃D
7.	(∃x) ~ (Gxa ⊃ ~ x = a)✓	3 ~ ∀D
8.	~ (Gba ⊃ ~ b = a)✓	7 ∃D
9.	Gac	4 ∃D
10.	Gba	8 ~ ⊃D
11.	~ ~ b = a	8 ~ ⊃D
12.	~ Gba	6 ∀D
	×	

The tree is closed. Therefore the sentence ‘[(∃x)Gax & ~ (∃x)Gxa] ⊃ (∀x)(Gxa ⊃ ~ x = a)’ is quantificationally true.

3.a. 1.	$\sim (\exists x)x = f(a)$	SM
2.	$(\forall x) \sim x = f(a)$	1 $\sim \exists$D
3.	$\sim f(a) = f(a)$	2 $\forall$D
	×	

The tree is closed. Therefore the given sentence is quantificationally true.

c. 1.	$\sim (\exists x)(\exists y)x = y$	SM
2.	$(\forall x) \sim (\exists y)x = y$	1 $\sim \exists$D
3.	$\sim (\exists y)a = y$	2 $\forall$D
4.	$(\forall y) \sim a = y$	3 $\sim \exists$D
5.	$\sim a = a$	4 $\forall$D
	×	

The tree is closed. Therefore the given sentence is quantificationally true.

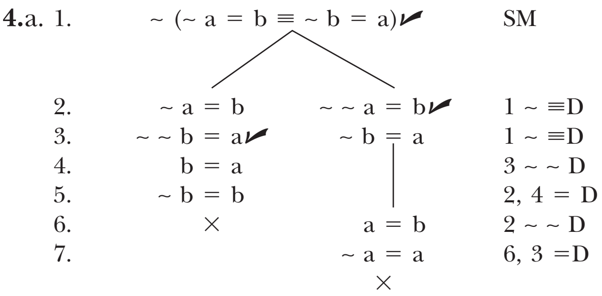
e. 1.	$\sim (\forall x)[Gx \supset (\exists y)f(x) = y]$	SM
2.	$(\exists x) \sim [Gx \supset (\exists y)f(x) = y]$	1 $\sim \forall$D
3.	$\sim [Ga \supset (\exists y)f(a) = y]$	2 $\exists$D
4.	Ga	3 $\sim \supset$D
5.	$\sim (\exists y)f(a) = y$	3 $\sim \supset$D
6.	$(\forall y) \sim f(a) = y$	5 $\sim \exists$D
7.	$\sim f(a) = f(a)$	7 $\forall$D
	×	

The tree is closed. Therefore the given sentence is quantificationally true.

g. 1.	$\sim (\forall y)\sim[(\forall x)x = y \vee (\forall x)f(x) = y]$ ✓		SM
2.	$(\exists y) \sim \sim [(\forall x)x = y \vee (\forall x)f(x) = y]$ ✓		1 $\sim \forall$D
3.	$\sim \sim [(\forall x)x = a \vee (\forall x)f(x) = a]$ ✓		2 $\exists$D
4.	$[(\forall x)x = a \vee (\forall x)f(x) = a]$ ✓		3 $\sim \sim \exists$D
5.	$(\forall x)x = a$	$(\forall x)f(x) = a$	4 $\vee$D
6.	$a = a$	$f(a) = a$	5 $\forall$D
	o		

The tree has a completed open branch. Therefore the given sentence is not quantificationally true.

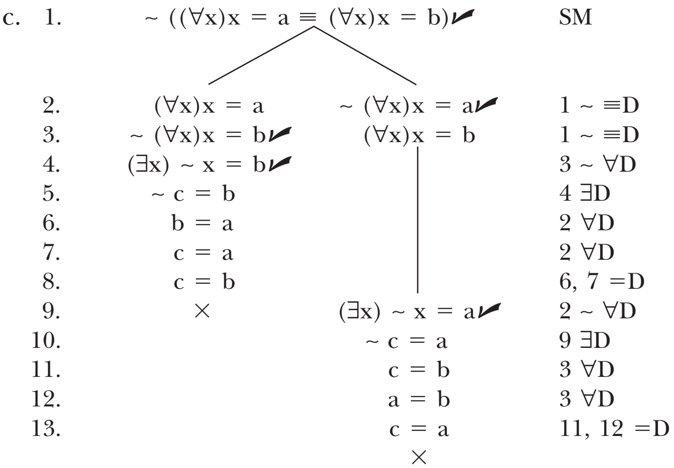
4.a.



1.	~ (~ a = b ≡ ~ b = a)✓		SM
2.	~ a = b	~ ~ a = b✓	1 ~ ≡D
3.	~ ~ b = a✓	~ b = a	1 ~ ≡D
4.	b = a		3 ~ ~ D
5.	~ b = b		2, 4 = D
6.	×	a = b	2 ~ ~ D
7.		~ a = a	6, 3 =D
		×	

The tree is closed. Therefore the sentences '~ a = b' and '~ b = a' are quantificationally equivalent.

c.



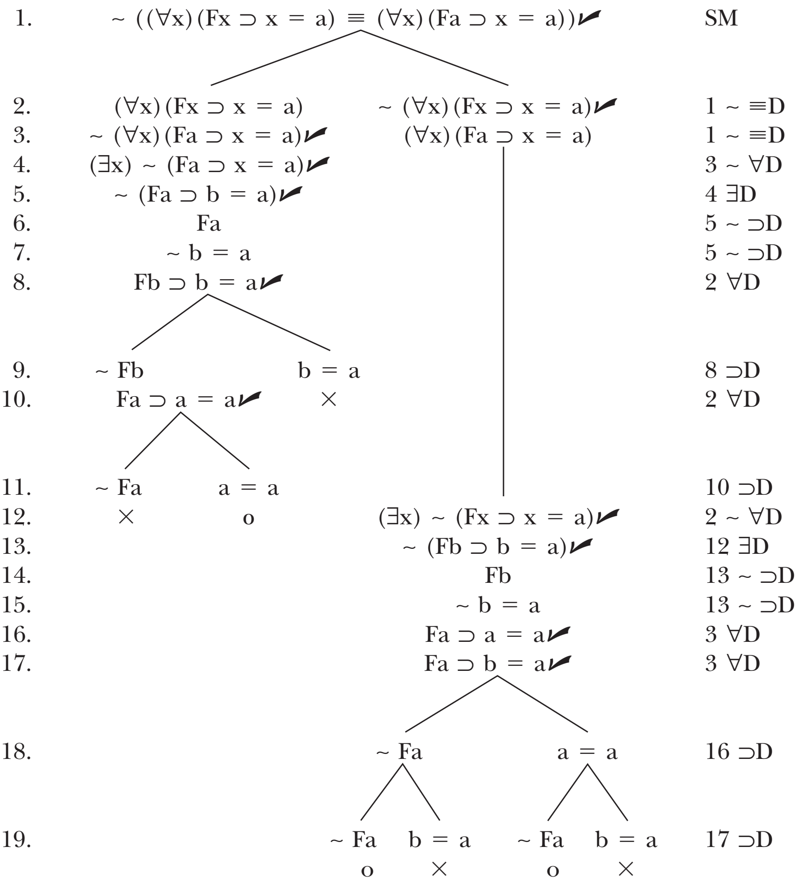
1.	~ ((∀x)x = a ≡ (∀x)x = b)✓		SM
2.	(∀x)x = a	~ (∀x)x = a✓	1 ~ ≡D
3.	~ (∀x)x = b✓	(∀x)x = b	1 ~ ≡D
4.	(∃x) ~ x = b✓		3 ~ ∀D
5.	~ c = b		4 ∃D
6.	b = a		2 ∀D
7.	c = a		2 ∀D
8.	c = b		6, 7 =D
9.	×	(∃x) ~ x = a✓	2 ~ ∀D
10.		~ c = a	9 ∃D
11.		c = b	3 ∀D
12.		a = b	3 ∀D
13.		c = a	11, 12 =D
		×	

The tree is closed. Therefore the sentences '(∀x)x = a' and '(∀x)x = b' are quantificationally equivalent.

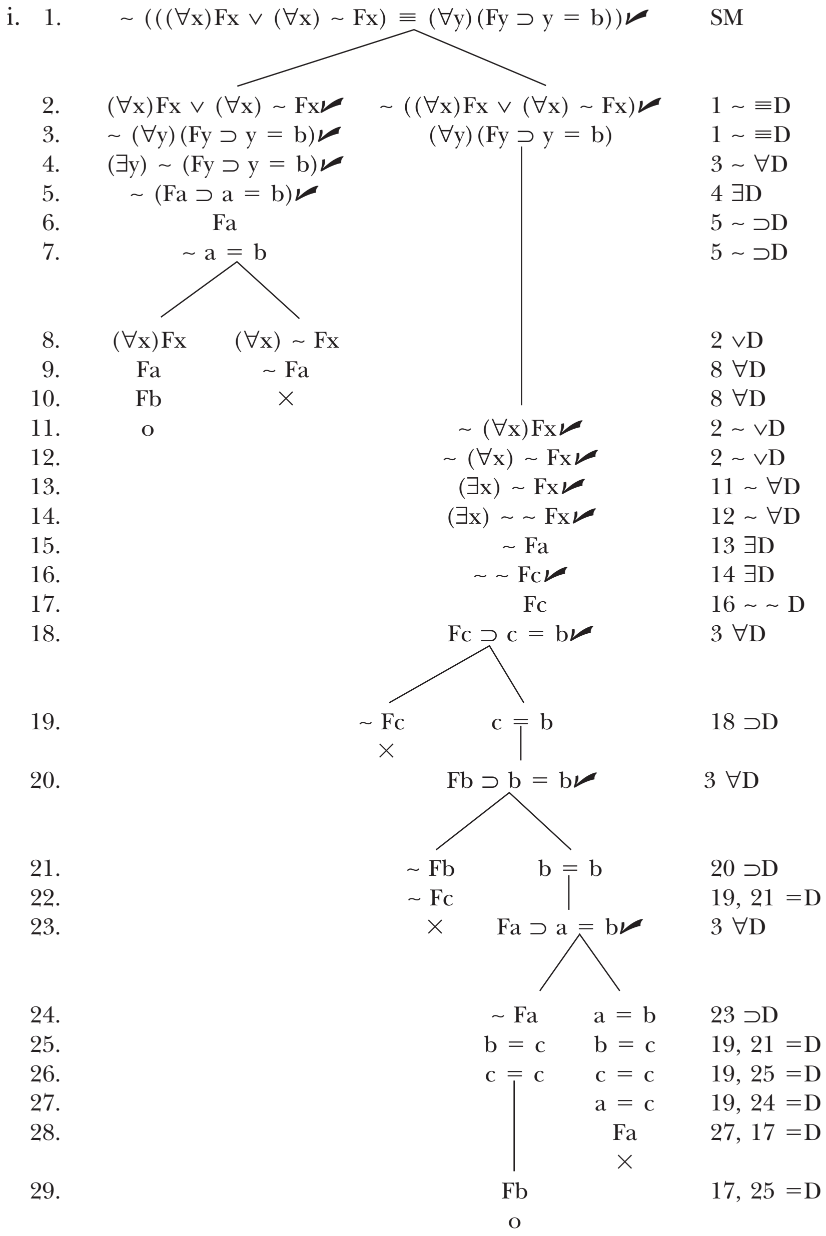
e.	1.	~ ((∀x)(∀y)x = y ≡ (∀x)x = a)✓		SM
	2.	(∀x)(∀y)x = y	~ (∀x)(∀y)x = y✓	1 ~ ≡D
	3.	~ (∀x)x = a✓	(∀x)x = a	1 ~ ≡D
	4.	(∃x) ~ x = a✓		3 ~ ∀D
	5.	~ b = a		4 ∃D
	6.	(∀y)b = y		2 ∀D
	7.	b = a		6 ∀D
	8.	×	(∃x) ~ (∀y)x = y✓	2 ~ ∀D
	9.		~ (∀y)b = y✓	8 ∃D
	10.		(∃y) ~ b = y✓	9 ~ ∀D
	11.		~ b = c	10 ∃D
	12.		b = a	3 ∀D
	13.		c = a	3 ∀D
	14.		b = c	12, 13 =D
			×	

The tree is closed. Therefore the sentences ‘(∀x)(∀y)x = y’ and ‘(∀x)x = a’ are quantificationally equivalent.

g.



The tree has at least one completed open branch, therefore the given sentences are not quantificationally equivalent.



The tree has at least one completed open branch, therefore the given sentences are not quantificationally equivalent.

```
k.  1.          ~ ((∃x)(x = a & x = b) ≡ a = b)✓            SM
                          /                  \
    2.   (∃x)(x = a & x = b)✓     ~ (∃x)(x = a & x = b)✓     1 ~ ≡D
    3.        ~ a = b                      a = b              1 ~ ≡D
    4.           |              (∀x) ~ (x = a & x = b)        2 ~ ∃D
    5.           |               ~ (a = a & a = b)✓           4 ∀D
    6.           |               ~ (b = a & b = b)            4 ∀D
                 |                     /        \
    7.           |                ~ a = a      ~ a = b        5 ~ &D
    8.    c = a & c = b✓             ×            ×           2 ∃D
    9.        c = a                                           8 &D
   10.        c = b                                           8 &D
   11.       ~ c = b                                          3, 9 =D
                ×
```

The tree is closed. Therefore the sentences '(∃x)(x = a & x = b)' and 'a = b' are quantificationally equivalent.

```
5.a. 1.    a = b & ~ Bab✓      SM
     2.     ~ ~ (∀x)Bxx✓       SM
     3.       (∀x)Bxx          2 ~ ~ D
     4.        a = b           1 &D
     5.        ~ Bab           1 &D
     6.          Bbb           3 ∀D
     7.          Bab           4, 6 =D
                  ×
```

The tree is closed. Therefore the argument is quantificationally valid.

```
c.  1.   (∀z)(Gz ⊃ (∀y)(Ky ⊃ Hzy))       SM
    2.      (Ki & Gj) & i = j✓            SM
    3.            ~ Hii                   SM
    4.           Ki & Gj✓                 2 &D
    5.             i = j                  2 &D
    6.              Ki                    4 &D
    7.              Gj                    4 &D
    8.    Gj ⊃ (∀y)(Ky ⊃ Hjy)✓            1 ∀D
               /          \
    9.   ~ Gj          (∀y)(Ky ⊃ Hjy)     8 ⊃D
   10.     ×             Ki ⊃ Hji✓        9 ∀D
                          /    \
   11.                 ~ Ki     Hji       10 ⊃D
   12.                   ×      Hii       5, 11 =D
                                 ×
```

The tree is closed. Therefore the argument is quantificationally valid.

e.	1.	a = b	SM
	2.	~ (Ka ∨ ~ Kb)✓	SM
	3.	~ Ka	2 ~ ∀D
	4.	~ ~ Kb✓	2 ~ ∀D
	5.	Kb	4 ~ ~ D
	6.	Ka	1, 5 =D
		×	

The tree is closed. Therefore the argument is quantificationally valid.

g.	1.	(∀x) (x = a ∨ x = b)		SM
	2.	(∃x) (Fxa & Fbx)✓		SM
	3.	~ (∃x)Fxx		SM
	4.	(∀x) ~ Fxx		3 ~ ∃D
	5.	Fca & Fbc✓		2 ∃D
	6.	Fca		5 &D
	7.	Fbc		5 &D
	8.	c = a ∨ c = b✓		1 ∀D
	9.			
	10.	c = a	c = b	8 ∨D
	11.	Fcc		6, 10 =D
	12.		Fcc	7, 10 =D
	13.	~ Fcc	~ Fcc	4 ∀D
		×	×	

The tree is closed. Therefore the argument is quantificationally valid.

i.	1.	(∀x) (∀y) (Fxy ∨ Fyx)		SM
	2.	a = b		SM
	3.	~ (∀x) (Fxa ∨ Fbx)✓		SM
	4.	(∃x) ~ (Fxa ∨ Fbx)✓		3 ~ ∀D
	5.	~ (Fca ∨ Fbc)✓		4 ∃D
	6.	~ Fca		5 ~ ∀D
	7.	~ Fbc		5 ~ ∀D
	8.	(∀y) (Fay ∨ Fya)		1 ∀D
	9.	Fac ∨ Fca✓		8 ∀D
	10.	Fac	Fca	9 ∀D
	11.	~ Fac	×	2, 7 =D
		×		

The tree is closed. Therefore the argument is quantificationally valid.

k. 1.	$(\forall x)(Fx \equiv {\sim} Gx)$		SM
2.	Fa		SM
3.	Gb		SM
4.	${\sim}{\sim}\, a = b$ ✓		SM
5.	$a = b$		$4\ {\sim}{\sim}D$
6.	$Fa \equiv {\sim} Ga$ ✓		$1\ \forall D$
7.	Fa	${\sim} Fa$	$6 \equiv D$
8.	${\sim} Ga$	${\sim}{\sim} Ga$	$6 \equiv D$
9.	Ga	×	$3, 5 = D$
	×		

The tree is closed. Therefore the argument is quantificationally valid.

m. 1.	$(\forall x)(\forall y)x = y$	SM
2.	${\sim}{\sim}(\exists x)(\exists y)(Fx \,\&\, {\sim} Fy)$ ✓	SM
3.	$(\exists x)(\exists y)(Fx \,\&\, {\sim} Fy)$ ✓	$2\ {\sim}{\sim}D$
4.	$(\exists y)(Fa \,\&\, {\sim} Fy)$ ✓	$3\ \exists D$
5.	$Fa \,\&\, {\sim} Fb$ ✓	$4\ \exists D$
6.	Fa	$5\ \&D$
7.	${\sim} Fb$	$5\ \&D$
8.	$(\forall y)a = y$	$1\ \forall D$
9.	$a = b$	$8\ \forall D$
10.	${\sim} Fa$	$7, 9 = D$
	×	

The tree is closed. Therefore the argument is quantificationally valid.

o. 1.	$(\forall x)(Hx \supset Hf(x))$		SM
2.	$(\exists z)\ {\sim} Hf(z)$		SM
3.	${\sim}{\sim}(\forall x)Hx$		SM
4.	$(\forall x)Hx$		$3\ {\sim}{\sim}D$
5.	${\sim} Hf(a)$		$2\ \exists D$
6.	Ha		$4\ \forall D$
7.	$Ha \supset Hf(a)$		$1\ \forall D$
8.	${\sim} Ha$	$Hf(a)$	$7 \supset D$
	×	×	

The tree is closed. Therefore the argument is quantificationally valid.

q.	1.	(∀x)(∀y)(Hxy ≡ ~ Hyx)		SM
	2.	(∃x)(Hx*f*(x) & ~ H*f*(x)x✓		SM
	3.	~ ~ (∀x)*f*(x) = x✓		SM
	4.	(∀x)*f*(x) = x		3 ~ ~ D
	5.	Ha*f*(a) & ~ H*f*(a)a✓		2 ∃D
	6.	Ha*f*(a)		5 &D
	7.	~ H*f*(a)a		5 &D
	8.	(∀y)(Hay ≡ ~ Hya)		1 ∀D
	9.	Haa ≡ ~ Haa✓		5 ∀D
	10.	Haa	~ Haa	9 ≡D
	11.	~ Haa	~ ~ Haa	9 ≡D
	12.	×	Haa	11 ~ ~ D
			×	

The tree is closed. Therefore the argument is quantificationally valid.

s.	1.	(∀x)[Px ⊃ (Ox v ~ x = *f*(b))]			SM
	2.	(∃x)[(Px & ~ Ox) & x = *f*(b)]✓			SM
	3.	~ Ob			SM
	4.	(Pa & ~ Oa) & a = *f*(b)✓			2 ∃D
	5.	Pa & ~ Oa✓			4 &D
	6.	a = *f*(b)			4 &D
	7.	Pa			5 &D
	8.	~ Oa			5 &D
	9.	Pa ⊃ (Oa v ~ a = *f*(b)✓			1 ∀D
	10.	Pb ⊃ (Ob v ~ b = *f*(b))			1 ∀D
	11.	~ Pa	Oa v ~ a = *f*(b)		9 ⊃D
		×			
			Oa	~ a = *f*(b)	11 ∨D
			×	×	

The tree is closed. Therefore the argument is quantificationally valid.

6.a.	1.		$(\forall x)(Fx \supset (\exists y)(Gyx \,\&\, \sim y = x))$	SM
	2.		$(\exists x)Fx$ ✓	SM
	3.		$\sim (\exists x)(\exists y) \sim x = y$ ✓	SM
	4.		$(\forall x) \sim (\exists y) \sim x = y$	3 $\sim \exists$D
	5.		Fa	2 $\exists$D
	6.		$Fa \supset (\exists y)(Gya \,\&\, \sim y = a)$ ✓	1 $\forall$D
	7.	$\sim Fa$	$(\exists y)(Gya \,\&\, \sim y = a)$ ✓	6 $\supset$D
	8.	×	$Gba \,\&\, \sim b = a$ ✓	7 $\exists$D
	9.		Gba	8 &D
	10.		$\sim b = a$	8 &D
	11.		$\sim (\exists y) \sim a = y$ ✓	4 $\forall$D
	12.		$\sim (\exists y) \sim b = y$ ✓	4 $\forall$D
	13.		$(\forall y) \sim \sim a = y$	11 $\sim \exists$D
	14.		$(\forall y) \sim \sim b = y$	12 $\sim \exists$D
	15.		$\sim \sim a = a$ ✓	13 $\forall$D
	16.		$\sim \sim a = b$ ✓	13 $\forall$D
	17.		$\sim \sim b = a$ ✓	14 $\forall$D
	18.		$\sim \sim b = b$ ✓	14 $\forall$D
	19.		$a = a$	15 $\sim \sim$ D
	20.		$a = b$	16 $\sim \sim$ D
	21.		$b = a$	17 $\sim \sim$ D
	22.		$b = b$	18 $\sim \sim$ D
	23.		$\sim b = b$	10, 21 =D
			×	

The tree is closed. Therefore the alleged entailment does hold.

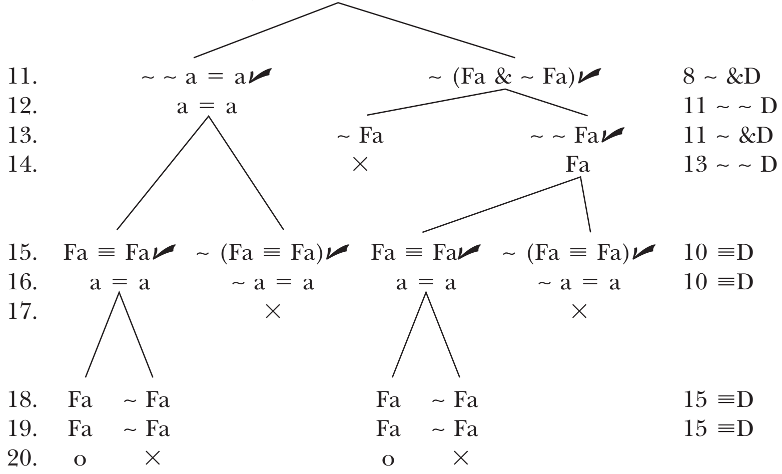
c.	1.		$(\forall x)(Fx \supset \sim x = a)$	SM
	2.		$(\exists x)Fx$ ✓	SM
	3.		$\sim (\exists x)(\exists y) \sim x = y$ ✓	SM
	4.		Fb	2 $\exists$D
	5.		$(\forall x) \sim (\exists y) \sim x = y$	3 $\sim \exists$D
	6.		$Fb \supset \sim b = a$ ✓	1 $\forall$D
	7.	$\sim Fb$	$\sim b = a$	6 $\supset$D
	8.	×	$\sim (\exists y) \sim a = y$ ✓	5 $\forall$D
	9.		$(\forall y) \sim \sim a = y$	8 $\sim \exists$D
	10.		$\sim \sim a = b$ ✓	9 $\forall$D
	11.		$a = b$	10 $\sim \sim$ D
	12.		$\sim a = a$	7, 11 =D
			×	

The tree is closed. Therefore the alleged entailment does hold.

e.	1.	$(\exists w)(\exists z) \sim w = z$ ✓	SM
	2.	$(\exists w)Hw$ ✓	SM
	3.	$\sim (\exists w) \sim Hw$ ✓	SM
	4.	$(\forall w) \sim \sim Hw$	3 $\sim \exists$D
	5.	$(\exists z) \sim a = z$ ✓	1 $\exists$D
	6.	Hb	2 $\exists$D
	7.	$\sim a = c$	5 $\exists$D
	8.	$\sim \sim Ha$ ✓	4 $\forall$D
	9.	$\sim \sim Hb$ ✓	4 $\forall$D
	10.	$\sim \sim Hc$ ✓	4 $\forall$D
	11.	Ha	8 $\sim \sim$ D
	12.	Hb	9 $\sim \sim$ D
	13.	Hc	10 $\sim \sim$ D
		o	

The tree has a completed open branch. Therefore, the alleged entailment does not hold.

g.	1.	$(\forall x)(\forall y)((Fx \equiv Fy) \equiv x = y)$	SM
	2.	$(\exists z)Fz$ ✓	SM
	3.	$\sim (\exists x)(\exists y)(\sim x = y \;\&\; (Fx \;\&\; \sim Fy))$ ✓	SM
	4.	$(\forall x) \sim (\exists y)(\sim x = y \;\&\; (Fx \;\&\; \sim Fy))$	3 $\sim \exists$D
	5.	Fa	2 $\exists$D
	6.	$\sim (\exists y)(\sim a = y \;\&\; (Fa \;\&\; \sim Fy))$ ✓	4 $\forall$D
	7.	$(\forall y) \sim (\sim a = y \;\&\; (Fa \;\&\; \sim Fy))$	6 $\sim \exists$D
	8.	$\sim (\sim a = a \;\&\; (Fa \;\&\; \sim Fa))$ ✓	7 $\forall$D
	9.	$(\forall y)((Fa \equiv Fy) \equiv a = y)$	1 $\forall$D
	10.	$(Fa \equiv Fa) \equiv a = a$ ✓	9 $\forall$D



The tree has at least one completed open branch. Therefore, the alleged entailment does not hold.

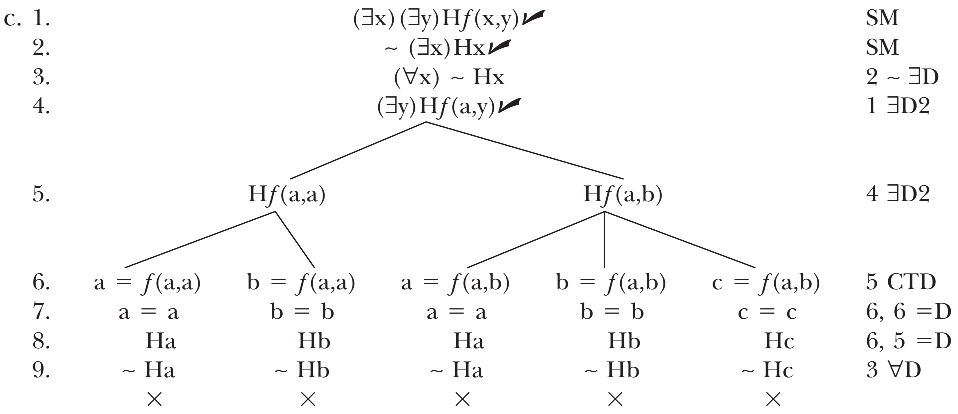
i.	1.	(∀x) (∀y) [~ x = g(y) ⊃ Gxy]		SM
	2.	~ (∃x) Gax✓		SM
	3.	~ (∃x)a = g(x)✓		SM
	4.	(∀x) ~ a = g(x)		3 ~ ∃D
	5.	~ a = g(a)		4 ∀D
	6.	(∀x) ~ Gax		2 ~ ∃D
	7.	(∀y) [~ a = g(y) ⊃ Gay]		1 ∀D
	8.	~ a = g(a) ⊃ Gaa✓		7 ∀D
	9.	~ Gaa		6 ∀D
	10.	~ ~ a = g(a)	Gaa	8 ⊃D
	11.	a = g(a)	×	10 ~ ~ D
		×		

The tree is closed. Therefore the entailment holds.

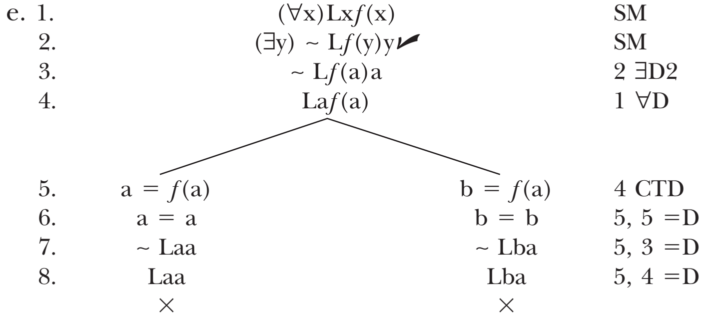
Section 9.6E

1. a.	1.	(∀x) (∀y) [~ x = g(y) ⊃ Gxy]			SM
	2.	~ (∃x) Gax✓			SM
	3.	(∀x) ~ Gax			2 ~ED
	4.	(∀y) [~ a = g(y) ⊃ Gay]			1 ∀D
	5.	~ Gaa			3 ∀D
	6.	~ a = g(a) ⊃ Gaa✓			4 ∀D
	7.	~ ~ a = g(a)✓		Gaa	6 ⊃D
	8.	a = g(a)		×	7 ~ ~ D
	9.	a = g(a)	b = g(a)		8 CTD
	10.	a = a	a = a		8, 8 =D
	11.	o	a = b		9, 8 =D
	12.		b = a		8, 9 =D

This systematic tree has a completed open branches. Therefore, the set being tested is quantificationally consistent.



This systematic tree is closed. Therefore, the set being tested is quantificationally inconsistent.



This systematic tree is closed. Therefore the set being tested is quantificationally inconsistent.

g.

1.	$(\forall x)(Gx \supset \sim Gh(x))$				SM
2.	$(\exists x)(\sim Gx \,\&\, \sim Gh(x))$ ✓				SM
3.	$\sim Ga \,\&\, \sim Gh(a)$ ✓				2 $\exists$D2
4.	$\sim Ga$				4 &D
5.	$\sim Gh(a)$				4 &D
6.	$a = h(a)$		$b = h(a)$		5 CTD
7.	$a = a$		$b = b$		6, 6 =D
8.	$\sim Ga$		$\sim Gb$		6, 5 =D
9.	$Ga \supset \sim Gh(a)$ ✓		$Ga \supset \sim Gh(a)$ ✓		1 $\forall$D
10.			$Gb \supset \sim Gh(b)$ ✓		1 $\forall$D
11.	$\sim Ga$	$\sim Gh(a)$	$\sim Ga$	$\sim Gh(a)$	9 $\supset$D
			⋮	⋮	

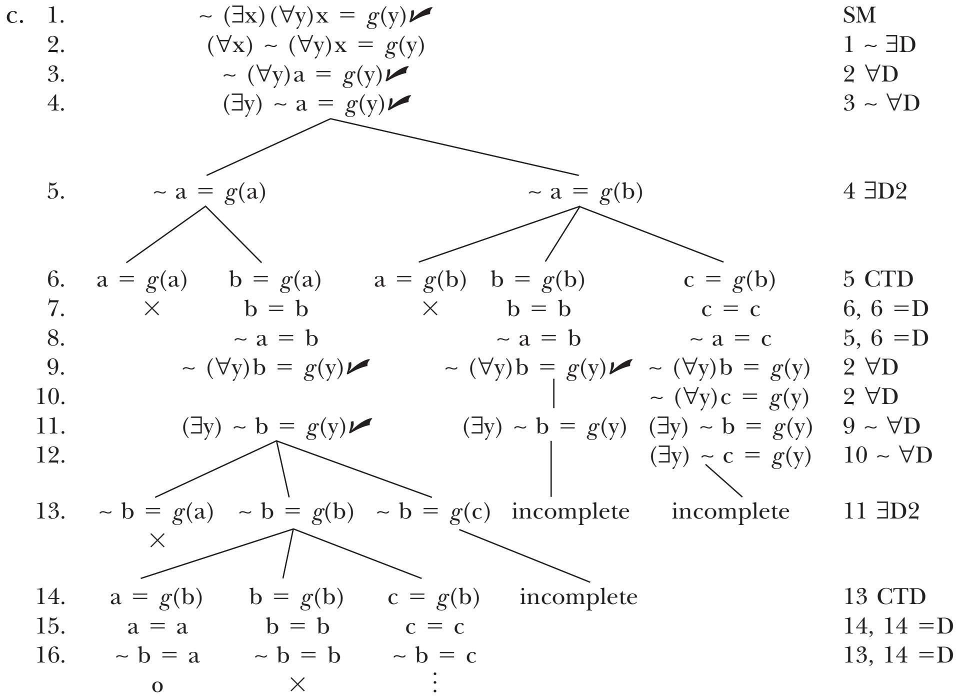
This systematic tree a completed open branches (in fact it has two, the left two). Therefore the set being tested is quantificationally consistent.

2.a.

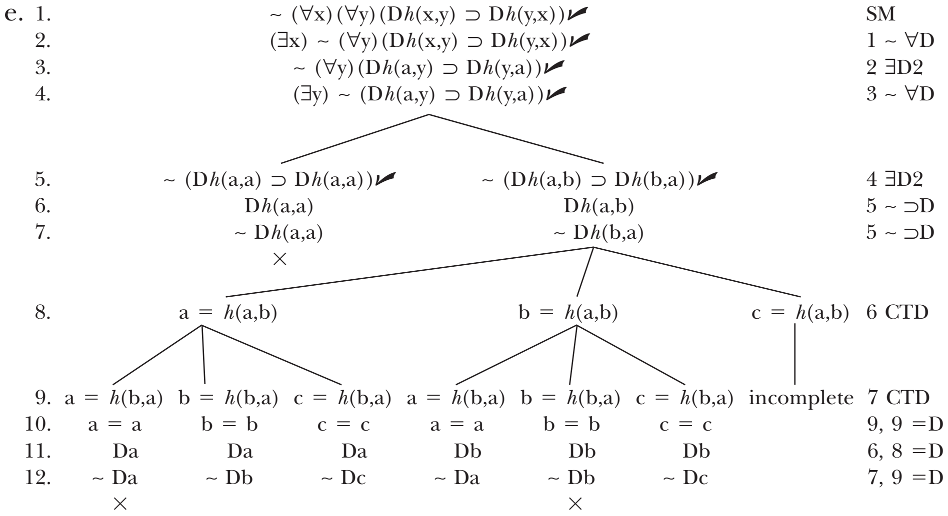
1.	$\sim (\forall x)(Pf(x) \supset Px)$ ✓		SM
2.	$(\exists x) \sim (Pf(x) \supset Px)$ ✓		1 ~ $\forall$D
3.	$\sim (Pf(a) \supset Pa)$ ✓		2 $\exists$D2
4.	$Pf(a)$		3 $\supset$D
5.	$\sim Pa$		3 $\supset$D
6.	$a = f(a)$	$b = f(a)$	4 CTD
7.	$a = a$	$b = b$	6, 6 =D
8.	Pa	Pb	4, 6 =D
	×	o	

The tree has a completed open branch. Therefore, the sentence being tested is not quantificationally false and the sentence of which it is the negation, '$(\forall x)(Pf(x) \supset Px)$' is not quantificationally true.

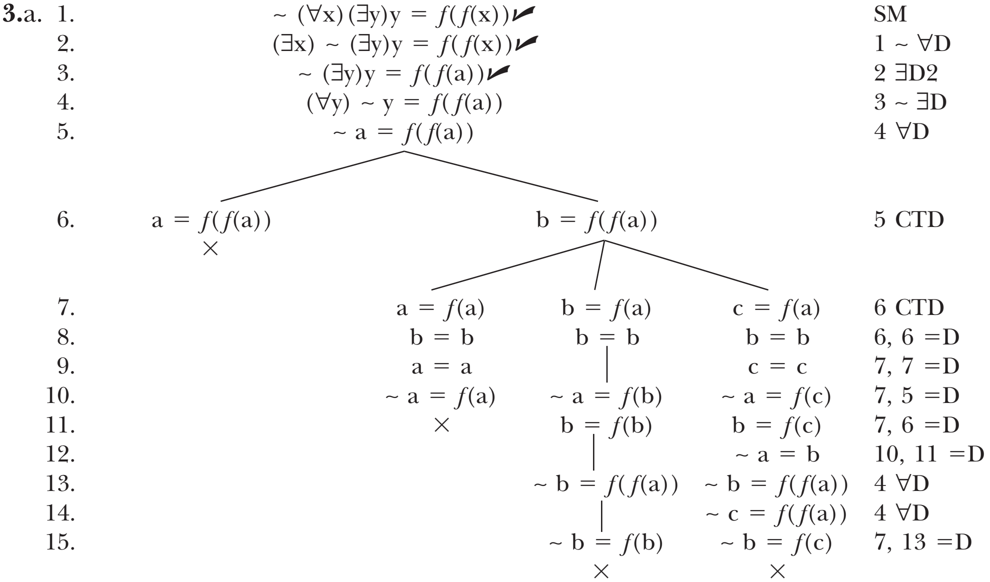
c.



If we were to complete the indicated missing work, we would have a systematic tree with at least one completed open branch (the left most branch). Therefore, the sentence being tested is not quantificationally false and the sentence of which it is a negation, '(∃x) (∀y)x = g(y)' is not quantificationally true.



If we were to complete the application of CTD and =D on the far right branch we would have a systematic tree with at least one completed open branch. Therefore, the sentence being tested is not quantificationally false, and the sentence of which it is the negation, '(∀x)(∀y)(Dh(x,y) ⊃ Dh(y,x))' is not quantificationally true.



The tree is closed. Therefore '~ (∀x)(∃y)y = f(f(x))' is quantificationally false and '(∀x)(∃y)y = f(f(x))' is quantificationally true.

c. 1.	$\sim [(\forall x)Lf(x) \supset (\forall x)Lf(f(x))]$ ✓	SM
2.	$(\forall x)Lf(x)$	$1 \sim \supset D$
3.	$\sim (\forall x)Lf(f(x))$ ✓	$1 \sim \supset D$
4.	$(\exists x) \sim Lf(f(x))$ ✓	$3 \sim \supset D$
5.	$\sim Lf(f(a))$	$4\ \exists D2$
6.	$Lf(a)$	$2\ \forall D$
7.	$\sim Lf(f(a))$	$2\ \forall D$
	×	

4.a. 1.	$\sim (\exists y) = f(f(x)$ ✓				SM
4.	$(\forall x) \sim f(x) = x$				$1 \sim \exists D$
3.	$\sim a = f(f(a))$				$4\ \forall D$
4.	$a = f(a)$	$b = f(a)$			3 CTD
5.	$\sim a = a$	$\sim b = a$			3, 4 =D
6.	×	$b = b$			4, 4 =D
7.		$\sim f(b) = b$			$2\ \forall D$
8.		$a = f(b)$	$b = f(b)$	$c = f(b)$	7 CTD
9.		$\sim a = b$	$\sim b = b$	$\sim c = b$	7, 8 =D
10.		$a = a$	×	$c = c$	8, 8 =D
		o			

. 1.	$\sim (\exists x)f(x) = x$ ✓		SM
2.	$f(a) = a$		$1\ \exists D2$
3.	$a = f(a)$	$b = f(b)$	2 CTD
4.	$a = a$	$b = b$	3, 3 =D
	o		

The tree for the negation of the given sentence has at least one completed open branch, and the tree for the given sentence has at least one completed open branch. Therefore the given sentence is quantificationally indeterminate.

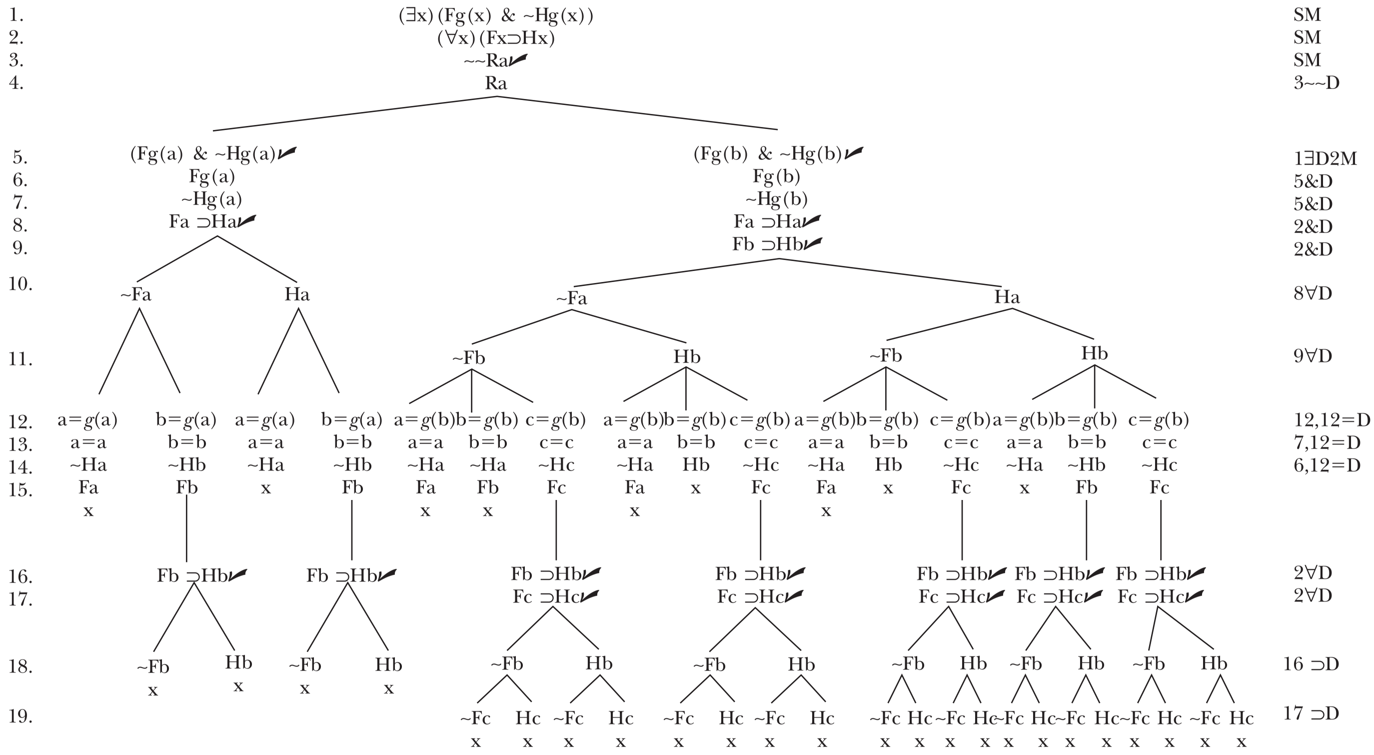
c. 1.	$(\forall x)(\exists y)y = f(f(x))$				SM
2.	$(\exists y)y = f(f(a))$ ✓				$1\ \forall D$
3.	$a = f(f(a))$		$b = f(f(a))$		$2\ \exists D2$
4.	$a = f(a)$	$b = f(a)$	$a = f(a)$	$b = f(a)$	3 CTD
5.	$a = a$	$b = b$	$a = a$	$b = b$	4, 4 =D
	o	⋮	⋮	⋮	

The tree has one completed open branch. Therefore the sentence '$(\forall x)(\exists y)y = f(f(x))$' is not quantificationally false.

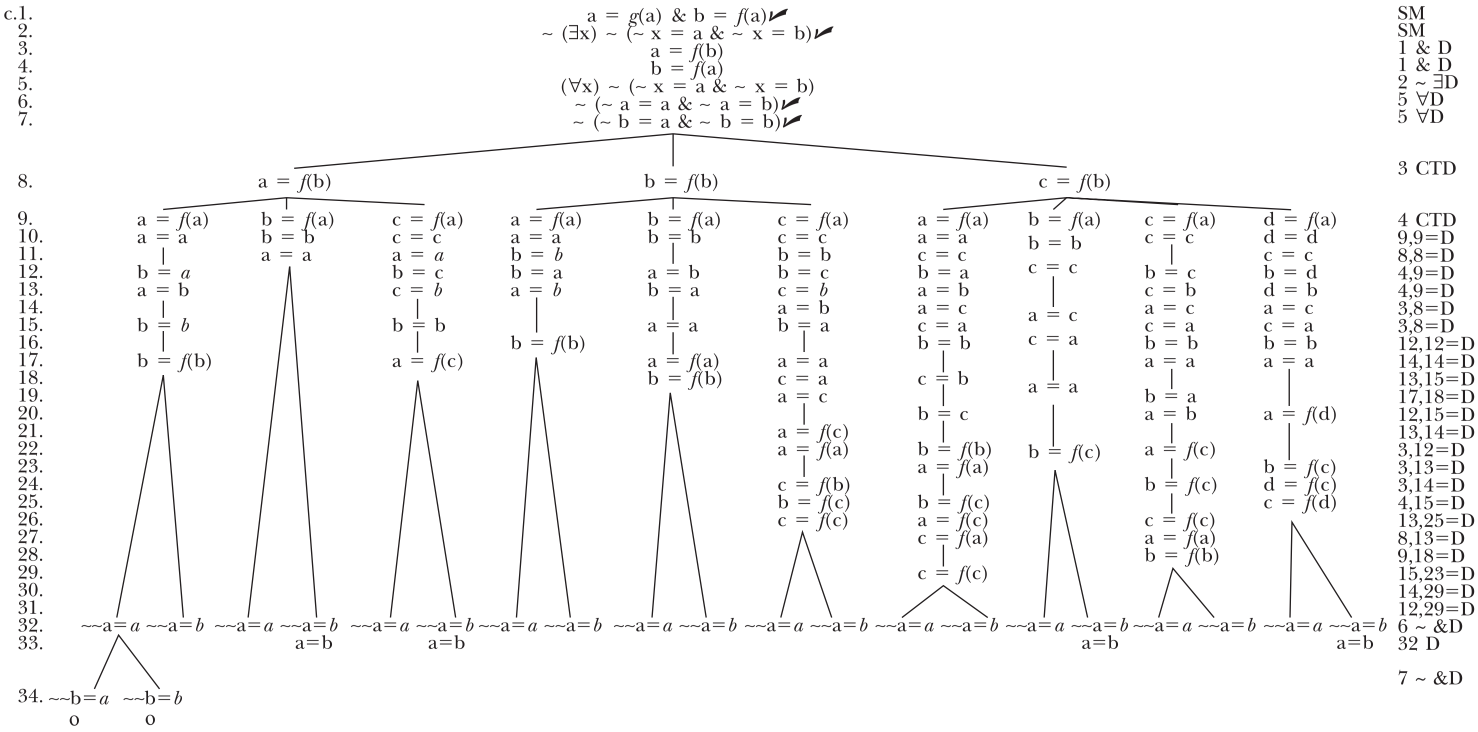
1.			$\sim (\forall x)(\exists y)y = f(f(x))$ ✓		SM
2.			$(\exists x) \sim (\exists y)y = f(f(x))$ ✓		$1 \sim \forall D$
3.			$\sim (\exists y)y = f(f(a))$ ✓		$2\ \exists D2$
4.			$(\forall y) \sim y = f(f(a))$		$3 \sim \exists D$
5.			$\sim a = f(f(a))$		$4\ \forall D$
6.	$a = f(a)$		$b = f(a)$		5 CTD
7.	$a = a$		$b = b$		6, 6 =D
8.	$\sim a = f(a)$		$\sim a = f(b)$		6, 5 =D
9.	×		$\sim b = f(f(a))$		$4\ \forall D$
10.		$a = f(b)$	$b = f(b)$	$c = f(b)$	8 CTD
11.		$a = a$		$c = c$	10, 10 =D
12.		$\sim a = a$	$\sim a = b$	$\sim a = c$	10, 8 =D
13.		×	$\sim b = f(b)$	$\sim b = f(b)$	6, 9 =D
14.			$\sim b = b$	$\sim b = c$	10, 13 =D
15.			×	$\sim c = f(f(a))$	$4\ \forall D$
16.				$\sim c = f(b)$	6, 15 =D
17.				$\sim c = c$	10, 16 =D
				×	

The tree is closed. Therefore '$\sim (\forall x)(\exists y)y = f(f(x))$' is quantificationally false and '$(\forall x)(\exists y)y = f(f(x))$' is quantificationally true.

5. a.

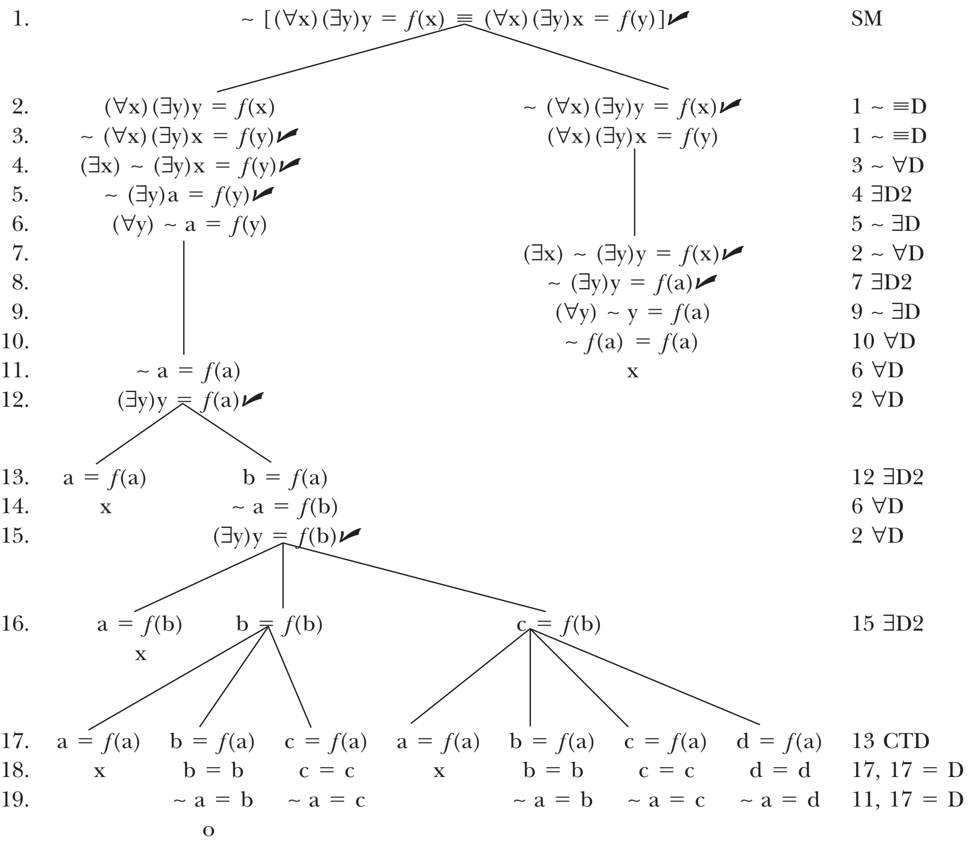


The tree for the premises and the negation of the conclusion is closed. Therefore, the argument being tested is quantificationally valid.



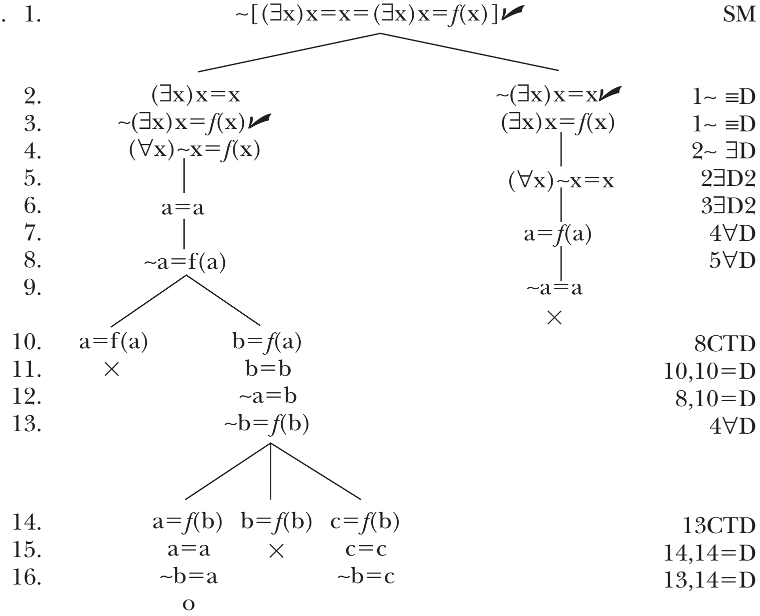
The tree has two completed open branches. Therefore, the argument is not quantificationally valid.

6. a.



The tree has a completed open branch. Therefore the sentences are not quantificationally equivalent.

c.



The systematic tree has at least one completed open branch. Therefore the sentences are not quantificationally equivalent.

7. a.

1.	(∀x)(∀y)x = g(x,y)	SM
2.	~ (∀x)x = g(x,x)✓	SM
3.	(∃x) ~ x = g(x,x)✓	2 ~ ∀D
4.	~ a = g(a,a)	3 ∃D2
5.	(∀y)a = g(a,y)	1 ∀D
6.	a = g(a,a)	1 ∀D
	×	

This tree is closed. Therefore, the alleged entailment does hold.

c. 1.	$\sim (\forall x)x = f(f(x))$			SM
2.	$\sim (\forall x)x = f(x) \checkmark$			SM
3.	$\sim (\exists x) \sim x = \mathrm{f}(a) \checkmark$			$2 \sim \forall D$
4.	$\sim a = f(a)$			$3\ \exists D$
5.	$a = f(f(a))$			$1\ \forall D$
6.	$a = f(a)$	$b = f(a)$		4 CTD
7.	×	$\sim a = b$		4, 6 =D
8.		$a = f(b)$		5, 6 =D
9.		$b = b$		6, 6 =D
10.		$a = a$		8, 8 =D
11.		$b = f(f(b))$		$1\ \forall D$
10.	$a = f(b)$	$b = f(b)$	$c = f(b)$	8 CTD
	o			

The tree has at least one completed one branch. Therefore the alleged entailment does not hold.

CHAPTER TEN

10.1 Derivability

1. a. Derive: (∀y)Fy

1	(∀x)Fx	Assumption
2	Fa	1 ∀E
3	(∀y)Fy	2 ∀I

c. Derive: (∃x)(∃y)Hxy

1	(∀x)(∀y)Hxy	Assumption
2	(∀y)Hay	1 ∀E
3	Hab	2 ∀E
4	(∃y)Hay	3 ∃I
5	(∃x)(∃y)Hxy	4 ∃I

e. Derive: Kg

1	(∀x)(∀y)Hxy	Assumption
2	Hab ⊃ Kg	Assumption
3	(∀y)Hay	1 ∀E
4	Hab	3 ∀E
5	Kg	2, 4 ⊃E

g. Derive: (∃y)Wy

1	(∀x)Sx	Assumption
2	(∃y)Sy ⊃ (∀w)Ww	Assumption
3	Sa	1 ∀E
4	(∃y)Sy	3 ∃I
5	(∀w)Ww	2, 4 ⊃E
6	Wa	5 ∀E
7	(∃y)Wy	6 ∃I

i. Derive: (∃x)(Lxx & Hxx)

1	(∀x)(∀y)Lxy	Assumption
2	(∃w)Hww	Assumption
3	Haa	A / ∃E
4	(∀y)Lay	1 ∀E
5	Laa	4 ∀E
6	Laa & Haa	3, 6 &I
7	(∃x)(Lxx & Hxx)	6 ∃I
8	(∃x)(Lxx & Hxx)	2, 3–7 ∃E

2. The mistakes in the attempted derivations are indicated and explained below.

a. Derive: Na

1	$(\forall x)Hx \supset \sim (\exists y)Ky$	Assumption	
2	$Ha \supset Na$	Assumption	
3	Ha	1 $\forall$E	**MISTAKE!**
4	Na	2, 3 $\supset$E	

Universal Elimination is a rule of inference. Like all rules of inference, it can be applied only to whole sentences, not to a formula or sentence that is a component of a larger sentence, and '$(\forall x)Hx$' is a component of the larger sentence, namely '$(\forall x)Hx \supset \sim (\exists y)Ky$.

c. Derive: $(\exists x)Cx$

1	$(\exists y)Fy$	Assumption	
2	$(\forall w)(Fw \equiv Cw)$	Assumption	
3	Fa	1 $\exists$E	**MISTAKE!**
4	$Fa \equiv Ca$	2 $\forall$E	
5	Ca	3, 4 $\equiv$E	
6	$(\exists x)Cx$	5 $\exists$I	

Existential Elimination is a rule that requires the construction of a subderivation. Here is a correctly done derivation:

Derive: $(\exists x)Cx$

1	$(\exists y)Fy$	Assumption
2	$(\forall w)(Fw \equiv Cw)$	Assumption
3	$\quad\vert\ Fa$	1 / $\exists$E
4	$\quad\vert\ Fa \equiv Ca$	2 $\forall$E
5	$\quad\vert\ Ca$	3, 4 $\equiv$E
6	$\quad\vert\ (\exists x)Cx$	5 $\exists$I
7	$(\exists x)Cx$	2, 3–6 $\exists$E

e. Derive: $(\exists y)(\forall x)Ayx$

1	$(\forall x)(\exists y)Ayx$	Assumption	
2	$(\forall x)Aax$	1 $\forall$E	**MISTAKE!**
3	$(\exists y)(\forall x)Ayx$	2 $\exists$I	

Universal Elimination takes us from a Universally quantified sentence to a substitution instance of that sentence. Here we start with a universally quantified sentence but instead of dropping the universal quantifier the existential quantifier, which comes after the universal quantifier, has been dropped. There is no correct derivation in this case. The sentence on line 3 is not derivable in *PD* from the sentence on line 1.

10.2E EXERCISE ANSWERS

1. Validity

a. Derive: $(\forall x)(Fx \supset Hx)$

1	$(\forall y)[Fy \supset (Gy \& Hy)]$	Assumption
2	Fc	A / $\supset$I
3	$Fc \supset (Gc \& Hc)$	1 $\forall$E
4	$Gc \& Hc$	2, 3 $\supset$E
5	Hc	4 &E
6	$Fc \supset Hc$	2–5 $\supset$I
7	$(\forall x)(Fx \supset Hx)$	6 $\forall$I

#c. Our derivation of the conclusion from the premises will use Universal Elimination, Existential Elimination, and Existential Introduction. We will make Existential Elimination our primary strategy:

Derive: $(\exists z)Fz$

1	$(\forall y)[Gy \supset (Hy \& Fy)]$	Assumption
2	$(\exists x)Gx$	Assumption
3	Ga	A / $\exists$E
G	$(\exists z)Fz$	
G	$(\exists z)Fz$	2, 3–__ $\exists$E

We will next use Universal Elimination to obtain a material conditional whose antecedent is ‘Ga’, allowing us to use Conditional Elimination to obtain ‘Ha & Fa’. The rest is straightforward:

Derive: $(\exists z)Fz$

1	$(\forall y)[Gy \supset (Hy \& Fy)]$	Assumption
2	$(\exists x)Gx$	Assumption
3	Ga	A / $\exists$E
4	$Ga \supset (Ha \& Fa)$	1 $\forall$E
5	$Ha \& Fa$	3, 4 $\supset$E
6	Fa	5 &E
7	$(\exists z)Fz$	6 $\exists$I
8	$(\exists z)Fz$	2, 3–7 $\exists$E

e. Derive: (∀x)Hx

1	(∃x)Fx ⊃ (∀x)Gx	Assumption
2	Fa	Assumption
3	(∀x)(Gx ⊃ Hx)	Assumption
4	(∃x)Fx	2 ∃I
5	(∀x)Gx	1, 4 ⊃E
6	Gb	5 ∀E
7	Gb ⊃ Hb	3 ∀E
8	Hb	6, 7 ⊃E
9	(∀x)Hx	8 ∀I

Note that it is essential that the constant chosen as the instantiating constant in line 6 be other than ‘a’, for ‘a’ occurs in an open assumption and were ‘a’ also used at line 6 we would violate the first restriction on Universal Introduction at line 9—for the instantiating constant, ‘a’, would then occur in an open assumption (on line 2).

g. Derive: (∀x)(Fx ∨ Gx)

1	(∀x)Fx ∨ (∀x)Gx	Assumption
2	(∀x)Fx	A / ∨E
3	Fa	2 ∀E
4	Fa ∨ Ga	3 ∨I
5	(∀x)Gx	A / ∨E
6	Ga	5 ∨E
7	Fa ∨ Ga	6 ∨I
8	Fa ∨ Ga	1, 2–4, 5–7 ∨E
9	(∀x)(Fx ∨ Gx)	8 ∀I

#i. Since the conclusion is a universally quantified sentence and there are no existentially quantified sentences among the premises, we will plan on deriving the conclusion by Universal Introduction and use Conditional Introduction to derive the substitution instance to which we will apply Universal Introduction:

Derive: (∀y)[(Fy ∨ Gy) ⊃ Hy]

1	(∀x)(Fx ⊃ Hx)	Assumption
2	(∀y)(Gy ⊃ Hy)	Assumption
3	Fb ∨ Gb	A / ⊃I
G	Hb	
G	(Fb ∨ Gb) ⊃ Hb	3–__ ⊃I
G	(∀y)[(Fy ∨ Gy) ⊃ Hy]	__ ∀I

Our plan will not violate the second restriction on Universal Introduction, for while the instantiating constant 'b' does occur in an assumption (at line 3), that assumpion will be closed at the point where we use Universal Introduction (the last line). The assumption on line 3 is a disjunction and we will now use Disjunction Elimination to obtain 'Hb'. To do so we will have to use Universal Elimination twice, once in association with each subderivation of the Disjunction Elimination strategy:

Derive: (∀y)[(Fy ∨ Gy) ⊃ Hy]

```
 1 | (∀x)(Fx ⊃ Hx)                      Assumption
 2 | (∀y)(Gy ⊃ Hx)                      Assumption
   |-------------------
 3 |  | Fa ∨ Ga                         A/ ⊃I
   |  |----------------
 4 |  |  | Fa                           A / ∨E
   |  |  |-------------
 5 |  |  | Fa ⊃ Ha                      1 ∀E
 6 |  |  | Ha                           4, 5 ⊃E
   |  |
 7 |  |  | Ga                           A / ∨E
   |  |  |-------------
 8 |  |  | Ga ⊃ Ha                      2 ∀E
 9 |  |  | Ha                           7, 8 ⊃E
10 |  | Ha                              3, 4–6, 7–9 ∨E
11 | (Fa ∨ Ga) ⊃ Ha                     3–10 ⊃I
12 | (∀y)[(Fy ∨ Gy) ⊃ Hy]               11 ∀I
```

k. Derive: (∀x)(Fx ⊃ Gx)

```
 1 | (∃x)Hx                             Assumption
2. | (∀x)(Hx ⊃ Rx)                      Assumption
3. | (∃x)Rx ⊃ (∀x)Gx                    Assumption
   |-------------------
 4 |  | Ha                              A / ∃E
   |  |----------------
 5 |  | Ha ⊃ Ra                         2 ∀E
 6 |  | Ra                              4, 5 ⊃E
 7 |  | (∃x)Rx                          6 ∃I
 8 |  | (∀x)Gx                          3, 7 ⊃E
 9 |  |  | Fb                           A / ⊃I
   |  |  |-------------
10 |  |  | Gb                           8 ∀E
11 |  | Fb ⊃ Gb                         9–10 ⊃I
12 |  | (∀x)(Fx ⊃ Gx)                   11 ∀I
13 | (∀x)(Fx ⊃ Gx)                      3, 4–12 ∃E
```

m. Derive: (∃y)(Hy ∨ Jy)

```
 1 | (∀x)Fx ∨ (∀y) ~ Gy              Assumption
 2 | Fa ⊃ Hb                         Assumption
 3 | ~ Gb ⊃ Jb                       Assumption
   |---------------------
 4 | | (∀x)Fx                        A / ∨E
   | |-------------------
 5 | | Fa                            4 ∀E
 6 | | Hb                            2, 5 ⊃E
 7 | | Hb ∨ Jb                       6 ∨I
 8 | | (∃y)(Hy ∨ Jy)                 7 ∃I
   |
 9 | | (∀y) ~ Gy                     A / ∨E
   | |-------------------
10 | | ~ Gb                          9 ∀E
11 | | Jb                            3, 10 ⊃E
12 | | Hb ∨ Jb                       11 ∨I
13 | | (∃y)(Hy ∨ Jy)                 12 ∃I
14 | (∃y)(Hy ∨ Jy)                   1, 4–8, 9–13 ∨E
```

2. Theorems

a. Derive: Fa ⊃ (∃y)Fy

```
 1  | | Fa                           A / ⊃I
    | |--------------
 2  | | (∃y)Fy                       1 ∃I
 3. | Fa ⊃ (∃y)Fy                    1–2 ⊃I
```

c. Derive: (∀x)[Fx ⊃ (Gx ⊃ Fx)]

```
 1 | | Fa                            A / ⊃I
   | |------------------
 2 | | | Ga                          A / ⊃I
   | | |----------------
 3 | | | Fa                          1 R
 4 | | Ga ⊃ Fa                       2–3 ⊃I
 5 | Fa ⊃ (Ga ⊃ Fa)                  1–4 ⊃I
 6 | (∀x)[Fx ⊃ (Gx ⊃ Fx)]            5 ∀I
```

e. Derive: ~ (∃x)Fx ⊃ (∀x) ~ Fx

```
 1 | | ~ (∃x)Fa                      A / ⊃I
   | |------------------
 2 | | | Fa                          A / ~ I
   | | |----------------
 3 | | | (∃x)Fx                      2 ∃I
 4 | | | ~ (∃x)Fx                    1 R
 5 | | ~ Fa                          2–4 ~ I
 6 | | (∀x) ~ Fx                     5 ∀I
 7 | ~(∃x)Fx ⊃ (∀x) ~ Fx             1–6 ⊃I
```

g. Derive: Fa ∨ (∃y) ~ Fy

1	│ │ ~ (Fa ∨ (∃y) ~ Fy)	A / ~ E
2	│ │ │ Fa	A / ~ I
3	│ │ │ Fa ∨ (∃y) ~ Fy	2 ∨I
4	│ │ │ ~ (Fa ∨ (∃y) ~ Fy	1 R
5	│ │ ~ Fa	2–4 ~ I
6	│ │ (∃y) ~ Fy	5 ∃I
7	│ │ Fa ∨ (∃y) ~ Fy	6 ∨I
8	│ │ ~ (Fa ∨ (∃y) ~ Fy)	1 R
9	│ Fa ∨ (∃y) ~ Fy	1–8 ~ E

#i. Since the theorem we want to prove is a material conditional, our primary strategy will be Conditional Introduction.

Derive: [(∀x)Fx ∨ (∀x)Gx] ⊃ (∀x)(Fx ∨ Gx)]

1	│ │ (∀x)Fx ∨ (∀x)Gx	A / ⊃I
G	│ │ (∀x)(Fx ∨ Gx)	
G	│ [(∀x)Fx ∨ (∀x)Gx] ⊃ (∀x)(Fx ∨ Gx)]	1–__ ⊃I

Our only accessible assumption is a disjunction, and our current goal is a universally quantified sentence. This suggests we will be using both Disjunction Elimination and Universal Introduction. The question is whether the goal of our Disjunction Elimination strategy should be '(∀x)(Fx ∨ Gx)' or a substitution instance of that sentence, say 'Fb ∨ Gb', with the intent of using Universal Introduction after we have used Disjunction Elimination. It turns out that both approaches will work. We will use the latter approach:

Derive: [(∀x)Fx ∨ (∀x)Gx] ⊃ (∀x)(Fx ∨ Gx)]

```
 1  | | (∀x)Fx ∨ (∀x)Gx                          A / ⊃I
    | |----------------------------
 2  | | | (∀x)Fx                                 A / ∨E
    | | |--------------------------
    | | |
 G  | | | Fb ∨ Gb

    | | | (∀x)Gx                                 A / ∨E
    | | |--------------------------
    | | |
 G  | | | Fb ∨ Gb
 G  | | Fb ∨ Gb                                  1, 2–__, __–__ ∨E
 G  | | (∀x)(Fx ∨ Gx)                            __ ∀I
 G  | [(∀x)Fx ∨ (∀x)Gx] ⊃ (∀x)(Fx ∨ Gx)]         1–__ ⊃I
```

Completing the two Disjunction Elimination subderivations is straightforward. In each case we will use Universal Elimination followed by Disjunction Introduction. To make this work we must, of course, in both cases use ‘b’ as our instantiating constant:

Derive: [(∀x)Fx ∨ (∀x)Gx] ⊃ (∀x)(Fx ∨ Gx)

```
 1  | | (∀x)Fx ∨ (∀x)Gx                          A / ⊃I
    | |----------------------------
 2  | | | (∀x)Fx                                 A / ∨E
    | | |--------------------------
 3  | | | Fb                                     2 ∀E
 4  | | | Fb ∨ Gb                                3 ∨I

 5  | | | (∀x)Gx                                 A / ∨E
    | | |--------------------------
 6  | | | Gb                                     5 ∀E
 7  | | | Fb ∨ Gb                                6 ∨I
 8  | | Fb ∨ Gb                                  1, 2–4, 5–7 ∨E
 9  | | (∀x)(Fx ∨ Gx)                            8 ∀I
10  | [(∀x)Fx ∨ (∀x)Gx] ⊃ (∀x)(Fx ∨ Gx)          1–9 ⊃I
```

Note that we could have done Universal Introduction within each of our innermost subderivations, thereby obtaining ‘(∀x)(Fx ∨ Gx)’ rather than ‘Fb ∨ Gb’ by Disjunction Elimination. Doing so would produce a derivation that is one line longer.

k. Derive: (∃x)(Fx & Gx) ⊃ [(∃x)Fx & (∃x)Gx]

```
 1  | | (∃x)(Fx & Gx)                            A / ⊃I
    | |----------------------------
 2  | | | Fa & Ga                                A / ∃E
    | | |--------------------------
 3  | | | Fa                                     2 &E
 4  | | | (∃x)Fx                                 3 ∃I
 5  | | | Ga                                     2 &E
 6  | | | (∃x)Gx                                 5 ∃I
 7  | | | (∃x)Fx & (∃x)Gx                        4, 6 &I
 8  | | (∃x)Fx & (∃x)Gx                          1, 2–7 ∃E
 9  | (∃x)(Fx &  Gx) ⊃ [(∃x)Fx & (∃x)Gx]         1–8 ⊃I
```

m. Derive: (∀x)Hx ≡ ~ (∃x) ~ Hx

1	│ │ (∀x)Hx	A / ≡I
2	│ │ │ (∃x) ~ Hx	A / ~ I
3	│ │ │ │ ~ Ha	A / ∃E
4	│ │ │ │ │ (∀x)Hx	A / ~ I
5	│ │ │ │ │ ~ Ha	3 R
6	│ │ │ │ │ Ha	1 ∀E
7	│ │ │ │ ~ (∀x)Hx	4–6 ~ I
8	│ │ │ ~ (∀x)Hx	2, 3–7 ∃E
9	│ │ │ (∀x)Hx	1 R
10	│ │ ~ (∃x) ~ Hx	2–9 ~ I
11	│ │ ~ (∃x) ~ Hx	A / ≡I
12	│ │ │ ~ Hb	A / ~ E
13	│ │ │ ~ (∃x) ~ Hx	11 R
14	│ │ │ (∃x) ~ Hx	12 ∃I
15	│ │ Hb	12–14 ~ E
16	│ │ (∀x)Hx	15 ∀I
17	│ (∀x)Hx ≡ ~ (∃x) ~ Hx	1–10, 11–16 ≡I

3. Equivalence

a. Derive: (∀x)Fx & (∀x)Gx

1	│ (∀x)(Fx & Gx)	Assumption
2	│ Fa & Ga	1 ∀E
3	│ Fa	2 &E
4	│ (∀x)Fx	3 ∀I
5	│ Ga	2 &E
6	│ (∀x)Gx	5 ∀I
7	│ (∀x)Fx & (∀x)Gx	4, 6 &I

Derive: (∀x)(Fx & Gx)

1	│ (∀x)Fx & (∀x)Gx	Assumption
2	│ (∀x)Fx	1 &E
3	│ Fa	2 ∀E
4	│ (∀x)Gx	1 &E
5	│ Ga	4 ∀E
6	│ Fa & Ga	3, 5 &I
7	│ (∀x)(Fx & Gx)	6 ∀I

c. Derive: ~ (∃x) ~ Fx

```
 1 | (∀x)Fx                          Assumption
   |----------------
 2 | | (∃x) ~ Fx                     A / ~I
   | |--------------
 3 | | | ~ Fa                        A / ∃E
   | | |------------
 4 | | | | (∀x)Fx                    A / ~ I
   | | | |----------
 5 | | | | Fa                        4 ∀E
 6 | | | | ~ Fa                      3 R
 7 | | | ~ (∀x)Fx                   4–6 ~ I
 9 | | ~ (∀x)Fx                      2, 3–7 ∃E
10 | | (∀x)Fx                        1 R
11 | ~ (∃x) ~ Fx                     2–10 ~ I
```

Derive: (∀x)Fx

```
1 | ~ (∃x) ~ Fx                      Assumption
  |----------------
2 | | ~ Fa                           A / ~ E
  | |--------------
3 | | (∃x) ~ Fx                      2 ∃I
4 | | ~ (∃x) ~ Fx                    1 R
5 | Fa                               2–4 ~ E
6 | (∀x)Fx                           5 ∀I
```

#e. Derive: ~ (∀x) ~ Fx

```
1 | (∃x)Fx                           Assumption
  |----------------
  |
  |
  |
G | ~ (∀x) ~ Fx
```

The one primary assumption of our derivation is an existentially quantified sentence, suggesting Existential Elimination as a possible strategy. The goal sentence is a negation, suggesting Negation Introduction. In fact, we will use both strategies, one within the other. In our first attempt we will use Existential Elimination as our primary strategy:

Derive: ~ (∀x) ~ Fx

```
1 | (∃x)Fx                           Assumption
  |----------------
2 | | Fa                             A / ∃E
  | |--------------
3 | | | (∀x) ~ Fx                    A / ~ I
  | | |------------
  | | |
  | | |
G | | ~ (∀x) ~ Fx                    3–__ ~I
G | ~ (∀x) ~ Fx                      1, 2–__ ∃E
```

We have taken '~ (∀x) ~ Fx' as our goal, within our Existential Elimination subderivation. Note that this sentence does not contain the constant 'a', so we are in no danger of violating the third restriction on Existential Elimination (that the instantiating constant not occur in the derived sentence). To complete the derivation we need to derive a sentence and its negation within the scope of the assumption on line 3. Only one negation is readily available, '~ Fa', which can be obtained by applying Universal Elimination to '(∀x) ~ Fx' on line 3. And 'Fa' can be obtained by Reiteration. So the completed derivation is

```
Derive: ~ (∀x) ~ Fx

1 │ (∃x)Fx                              Assumption
  ├──────────────
2 │ │ Fa                                A / ∃E
  │ ├────────────
3 │ │ │ (∀x) ~ Fx                       A / ~ I
  │ │ ├──────────
4 │ │ │ ~ Fa                            3 ∀E
5 │ │ │ Fa                              2 R
6 │ │ ~ (∀x) ~ Fx                       3–5 ~ I
7 │ ~ (∀x) ~ Fx                         1, 2–6 ∃E
```

To avoid violating the third restriction on Existential Elimination it is a good idea, at the time an Existential Elimination subderivation is started, to select the goal of that subderivation; making sure that the goal sentence does not contain the instantiating constant in the subderivation's assumption. In a derivation that uses Existential Elimination as its primary strategy the sentence that occurs on the last line should also appear as the last sentence in the subderivation. In this example that sentence is '~ (∀x) ~ Fx'.

To complete our demonstration that '(∃x)Fx' and '~ (∀x) ~ Fx' are equivalent we will now derive the first sentence from the second:

```
Derive: (∃x)Fx

1 │ ~ (∀x) ~ Fx                         Assumption
  ├──────────────
  │
  │
  │
  │
  │
  │
G │ (∃x)Fx
```

Here our goal sentence is an existentially quantified sentence, and our one primary assumption a negation. The former suggests Existential Introduction as a strategy, the latter suggests Negation Elimination (since we do have a negation readily available). We will construct two derivations to illustrate that both strategies work as the primary strategy, in each case sing the order strategy as a secondary strategy:

Derive: (∃x)Fx

1	│ ~ (∀x) ~ Fx	Assumption
2	│ │ ~ (∃x)Fx	A / ~ E
G	│ │ (∀x) ~ Fx	
	│ │ ~ (∀x) ~ Fx	1 R
G	│ (∃x)Fx	2–___ ~ E

We have decided to use '(∀x) ~ Fx' and '~ (∀x) ~ Fx' as the sentence and negation we derive for Negation Elimination. (We could of course, also have decided to use '(∃x)Fx' and '~ (∃x)Fx'.) Our current goal is '(∀x) ~ Fx', a universally quantified sentence. One way to obtain it is by Universal Introduction, which will require obtaining a substitution instance of that sentence. In planning for Universal Introduction we pick as our goal a substitution instance of the desired universally quantified sentence, and the instantiating constant in this substitution instance should not occur in any open assumption. Because neither of our assumptions contains a constant, we are free to choose any constant. We choose the substitution instance '~ Fa'. And since this sentence is a negation, we will try to obtain it by Negation Introduction:

Derive: (∃x)Fx

1	│ ~ (∀x) ~ Fx	Assumption
2	│ │ ~ (∃x)Fx	A / ~ E
3	│ │ │ Fa	A / ~ I
G	│ │ ~ Fa	3–___ ~ I
G	│ │ (∀x) ~ Fx	___ ∀I
	│ │ ~ (∀x) ~ Fx	1 R
G	│ (∃x)Fx	2–___ ~ E

As of line 3 two negations are available to us, '~ (∀x) ~ Fx' and '~ (∃x)Fx'. We select the latter to use within the negation strategy that begins at line 3 because the unnegated ‘(∃x)Fx’ is easily obtainable from line 3 by Existential Introduction:

Derive: (∃x)Fx

1	~ (∀x) ~ Fx	Assumption
2	~ (∃x)Fx	A / ~E
3	Fa	A / ~ I
4	(∃x)Fx	3 ∃I
5	~ (∃x)Fx	2 R
6	~ Fa	3–5 ~ I
7	(∀x) ~ Fx	6 ∀I
8	~ (∀x) ~ Fx	1 R
9	(∃x)Fx	2–8 ~ E

We have now derived each member of our original pair of sentences from the other, so we have demonstrated that these sentences, ‘(∃x)Fx’ and ‘~ (∀x) ~ Fx’ are equivalent in *PD*.

g. Derive: ~ (∃y)(Hy & Iy)

1	(∀z)(Hz ⊃ ~ Iz)	Assumption
2	(∃y)(Hy & Iy)	A / ~ I
3	Hb & Ib	A / ∃E
4	(∀z)(Hz ⊃ ~ Iz)	A / ~ I
5	Hb ⊃ ~ Ib	1 ∀E
6	Hb	3 &E
7	~ Ib	5, 6 ⊃E
8	Ib	3 &E
9	~ (∀z)(Hz ⊃ ~ Iz)	4–8 ~ I
10	~ (∀z)(Hz ⊃ ~ Iz)	2, 3–9 ∃E
11	(∀z)(Hz ⊃ ~ Iz)	1 R
12	~ (∃y)(Hy & Iy)	2–11 ~ I

Derive: (∀z)(Hz ⊃ ~ Iz)

1	~ (∃y)(Hy & Iy)	Assumption
2	Ha	A / ⊃I
3	Ia	A / ~ I
4	Ha & Ia	2, 3 &I
5	(∃y)(Hy & Iy)	4 ∃I
6	~ (∃y)(Hy & Iy)	1 R
7	~ Ia	3–6 ⊃I
8	Ha ⊃ ~ Ia	2–7 ⊃I
9	(∀z)(Hz ⊃ ~ Iz)	8 ∀I

i. Derive: $(\forall x)(Fx \supset (\exists y)Gy)$

1	│ $(\forall x)(\exists y)(Fx \supset Gy)$	Assumption
2	│ $(\exists y)(Fa \supset Gy)$	1 $\forall$E
3	│ │ $Fa \supset Gb$	A / $\exists$E
4	│ │ │ Fa	A / $\supset$I
5	│ │ │ Gb	3, 4 $\supset$I
6	│ │ │ $(\exists y)Gy$	5 $\exists$I
7	│ │ $Fa \supset (\exists y)Gy$	4–6 $\supset$I
8	│ $Fa \supset (\exists y)Gy$	2, 3–7 $\exists$E
9	│ $(\forall x)(Fx \supset (\exists y)Gy)$	8 $\forall$I

Derive: $(\forall x)(\exists y)(Fx \supset Gy)$

1	│ $(\forall x)(Fx \supset (\exists y)Gy)$	Assumption
2	│ │ $\sim (\exists y)(Fa \supset Gy)$	A / $\sim$ E
3	│ │ │ Fa	A / $\supset$I
5	│ │ │ $Fa \supset (\exists y)Gy$	1 $\forall$E
6	│ │ │ $(\exists y)Gy$	3, 5 $\supset$E
7	│ │ │ │ Gc	A / $\exists$E
8	│ │ │ │ │ $\sim Gb$	A / $\sim$ E
9	│ │ │ │ │ │ Fa	A / $\supset$I
10	│ │ │ │ │ │ Gc	7 R
11	│ │ │ │ │ $Fa \supset Gc$	9–10 $\supset$I
12	│ │ │ │ │ $(\exists y)(Fa \supset Gy)$	11 $\exists$I
13	│ │ │ │ │ $\sim (\exists y)(Fa \supset Gy)$	2 R
14	│ │ │ │ Gb	8–13 $\sim$ E
15	│ │ │ Gb	6, 7–14 $\exists$E
16	│ │ $Fa \supset Gb$	3–15 $\supset$I
17	│ │ $(\exists y)(Fa \supset Gy)$	16 $\exists$I
18	│ │ $\sim (\exists y)(Fa \supset Gy)$	2 R
19	│ $(\exists y)(Fa \supset Gy)$	2–18 $\exists$E
20	│ $(\forall x)(\exists y)(Fx \supset Gy)$	19 $\forall$I

4. Inconsistency

a. Derive: Fa, ~ Fa

1	│ $(\forall x)(Fx \equiv \sim Fx)$	Assumption
2	│ $Fa \equiv \sim Fa$	1 $\forall$E
3	│ │ Fa	A / $\sim$ I
4	│ │ $\sim Fa$	2, 3 $\equiv$E
5	│ │ Fa	3 R
6	│ $\sim Fa$	3–5 $\sim$ I
7	│ Fa	2, 6 $\equiv$E

#c. It is fairly easy to see that the set $\{\sim (\forall x)Fx, \sim (\exists x) \sim Fx\}$ is inconsistent. If not everything is F, then there must be something that is not F, but this contradicts the claim that there is not something that is not F. The set contains two negations. We choose to use one of them, ‘$\sim (\forall x)Fx$’, as $\sim Q$. Our derivation starts thus:

Derive: $(\forall x)Fx$, $\sim (\forall x)Fx$

1	$\sim (\forall x)Fx$	Assumption
2	$\sim (\exists x) \sim Fx$	Assumption
G	$(\forall x)Fx$	
	$\sim (\forall x)Fx$	1 R

How we should continue is not immediately clear. We reason as follows: The sentences that are accessible include only two negations. There is no rule of inference that can be applied to a negation to yield a further sentence (Negation Elimination starts with the auxiliary assumption of a negation, not with a primary assumption that is a negation.) So working from the “top down” is not here promising. Our current goal is a universally quantified sentence, and Universal Introduction is the rule that yields such sentences. So we will plan on using Universal Introduction. To use it, we must first derive a substitution instance of our goal sentence. Since there are no constants in the primary assumptions, which substitution instance doesn’t matter. We pick ‘Fa’.

Derive: $(\forall x)Fx$, $\sim (\forall x)Fx$

1	$\sim (\forall x)Fx$	Assumption
2	$\sim (\exists x) \sim Fx$	Assumption
G	Fa	
G	$(\forall x)Fx$	__ ∀I
	$\sim (\forall x)Fx$	1 R

The task now is to derive ‘Fa’. We have added to new assumptions, so working from the “top down” is still not promising. So we will try to get ‘Fa’ by Negation Elimination:

Derive: (∀x)Fx, ~ (∀x)Fx

1	~ (∀x)Fx	Assumption
2	~ (∃x) ~ Fx	Assumption
3	~ Fa	A / ~ E
G	Fa	
G	(∀x)Fx	__ ∀I
	~ (∀x)Fx	1 R

With our new assumption, we can now work from the “top down”. More specifically, we have ‘~ (∃x) ~ Fx’ at line 2 and from line 3 we can obtain, by Existential Introduction, ‘(∃x) ~ Fx’, giving us the **Q** and ~ **Q** we need to complete our Negation Elimination strategy and the derivation:

Derive: (∀x)Fx, ~ (∀x)Fx

1	~ (∀x)Fx	Assumption
2	~ (∃x) ~ Fx	Assumption
3	~ Fa	A / ~ E
4	(∃x) ~ Fx	3 ∃I
5	~ (∃x) ~ Fx	2 R
6	Fa	3–5 ~ E
7	(∀x)Fx	6 ∀I
8	~ (∀x)Fx	1 R

Our demonstration of inconsistency in PD is now complete. We have used Universal Introduction and met both restrictions on that rule: the instantiating constant ‘a’ does not occur in the sentence derived by Universal Introduction and it does not occur, as of line 7, in any open assumption.

e. Derive: (∃x)Gx, ~ (∃x)Gx

1	(∀x)(Fx ⊃ Gx)	Assumption
2	(∃x)Fx	Assumption
3	~ (∃x)Gx	Assumption
4	Fb	A / ∃E
5	Fb ⊃ Gb	1 ∀E
6	Gb	4, 5 ⊃E
7	(∃x)Gx	6 ∃I
8	(∃x)Gx	2, 4–7 ∃E
9	~ (∃x)Gx	3 R

g. Derive: (∀x)Fx, ~ (∀x)Fx

1	(∀x)Fx	Assumption
2	(∃y) ~ Fy	Assumption
3	~ Fa	A / ∃E
4	(∀x)Fx	A / ~ I
5	Fa	1 ∀E
6	~ Fa	3 R
7	~ (∀x)Fx	4–6 ~ I
8	~ (∀x)Fx	2, 3–7 ∃E
9	(∀x)Fx	1 R

i. Derive: (∀x)Fx, ~ (∀x)Fx

1	(∀x)(Hx ≡ ~ Gx)	Assumption
2	(∃x)Hx	Assumption
3	(∀x)Gx	Assumption
4	Hc	A / ∃E
5	(∀x)Gx	A / ~ I
6	Hc ≡ ~ Gc	1 ∀E
7	~ Gc	4, 6 ≡E
8	Gc	3 ∀E
9	~ (∀x)Gx	5–8 ~ I
10	~ (∀x)Gx	2, 4–9 ∃E
11	(∀x)Gx	3 R

k. Derive: (∃y)(Ry & My), ~ (∃y)(Ry & My)

1	│ (∀z)[Rz ⊃ (Tz & ~ Mz)]	Assumption
2	│ (∃y)(Ry & My)	Assumption
3	│ │ Ra & Ma	A / ∃E
4	│ │ │ (∃y)(Ry & My)	A / ~ I
5	│ │ │ Ra ⊃ (Ta & ~ Ma)	1 ∀E
6	│ │ │ Ra	3 &E
7	│ │ │ Ta & ~ Ma	5, 6 ⊃E
8	│ │ │ ~ Ma	7 &E
9	│ │ │ Ma	3 &E
10	│ │ ~ (∃y)(Ry & My)	4–9 ~ I
11	│ ~ (∃y)(Ry & My)	2, 3–10 ∃E
12	│ (∃y)(Ry & My)	2 R

5. Derivability

a. Derive: (∀x)(∃y)Fxy

1	│ (∃y)(∀x)Fxy	Assumption
2	│ │ (∀x)Fxa	A / ∃E
3	│ │ Fba	2 ∀E
4	│ │ (∃y)Fby	3 ∃I
5	│ (∃y)Fby	1, 2–3 ∃E
6	│ (∀x)(∃y)Fxy	5 ∀I

c. Derive: (∃x)(∃y)(∃z)Fxyz

1	│ (∃x)Fxxx	Assumption
2	│ │ Faaa	A / ∃E
3	│ │ (∃z)Faaz	2 ∃I
4	│ │ (∃y)(∃z)Fayz	3 ∃I
5	│ │ (∃x)(∃y)(∃z)Fxyz	4 ∃I
6	│ (∃x)(∃y)(∃z)Fxyz	1, 2–5 ∃E

e. Derive: (∃x)(∃y)Gyx

1	│ (∀x)(Fx ⊃ (∃y)Gxy)	Assumption
2	│ (∃x)Fx	Assumption
3	│ │ Fa	A / ∃E
4	│ │ Fa ⊃ (∃y)Gay	1 ∀E
5	│ │ (∃y)Gay	3, 4 ⊃E
6	│ │ │ Gab	A /∃E
7	│ │ │ (∃y)Gyb	6 ∃I
8	│ │ │ (∃x)(∃y)Gyx	7 ∃I
9	│ │ (∃x)(∃y)Gyx	5, 6–8 ∃E
10	│ (∃x)(∃y)Gyx	2, 3–9 ∃E

g. Derive: (∃x)(∃y) ~ Hyx

1	(∀x)(∀y)(Hxy ⊃ ~ Hyx)	Assumption
2	(∃x)(∃y)Hxy	Assumption
3	(∃y)Hxa	A / ∃E
4	Hba	A / ∃E
5	(∀y)(Hby ⊃ ~ Hyb)	1 ∀E
6	Hba ⊃ ~ Hab	5 ∀E
7	~ Hab	4, 6 ⊃E
8	(∃y) ~ Hyb	7 ∃I
9	(∃x)(∃y) ~ Hyx	8 ∃I
10	(∃x)(∃y) ~ Hyx	3, 4–9 ∃E
11	(∃x)(∃y) ~ Hyx	2, 3–10 ∃E

i. Derive: (∀x)(∀y)Hxy

1	~ (∃x)(∃y)Rxy	Assumption
2	(∀x)(∀y)(~ Hxy ≡ Rxy)	Assumption
3	~ Hab	A / ~ E
4	(∀y)(~ Hay ≡ Ray)	2 ∀E
5	~ Hab ≡ Rab	4 ∀E
6	Rab	3, 5 ≡E
7	(∃y)Ray	6 ∃I
9	(∃x)(∃y)Rxy	7 ∃I
10	~ (∃x)(∃y)Rxy	1 R
11	Hab	3–10 ~ E
12	(∀y)Hay	11 ∀I
13	(∀x)(∀y)Hxy	12 ∀I

6. Validity

a. Derive: (∃y)Gya

1	(∀x)(Fx ⊃ Gba)	Assumption
2	(∃x)Fx	Assumption
3	Fb	A / ∃E
4	Fb ⊃ Gba	1 ∀E
5	Gba	3, 4 ⊃E
6	(∃y)Gya	5 ∃I
7	(∃y)Gya	2, 3–6 ∃E

c. Derive: (∃x)(∃y)Fxy

1	│ (∃x)(∃y)(Fxy ∨ Fyx)	Assumption
2	│ │ (∃y)(Fay ∨ Fya)	A / ∃E
3	│ │ │ Fab ∨ Fba	A / ∃E
4	│ │ │ │ Fab	A / ∨E
5	│ │ │ │ (∃y)Fay	4 ∃I
6	│ │ │ │ (∃x)(∃y)Fxy	5 ∃I
7	│ │ │ │ Fba	A /∨E
8	│ │ │ │ (∃y)Fby	7 ∃I
9	│ │ │ │ (∃x)(∃y)Fxy	8 ∃I
10	│ │ │ (∃x)(∃y)Fxy	3, 4–6, 7–9 ∨E
11	│ │ (∃x)(∃y)Fxy	2, 3–10 ∃E
12	│ (∃x)(∃y)Fxy	1, 2–11 ∃E

e. Derive: (∀z)(Faz ⊃ Fza)

1	│ (∀x)(∀y)[(∃z)[(Fyz & ~ Fzx) ⊃ Gxy]	Assumption
2	│ ~ (∃x)Gxx	Assumption
3	│ │ Fab	A / ⊃I
4	│ │ │ ~ Fba	A / ~ E
5	│ │ │ (∀y)[(∃z)(Fyz & ~ Fza) ⊃ Gay]	1 ∀E
6	│ │ │ (∃z)(Faz & ~ Fza) ⊃ Gaa	5 ∀E
7	│ │ │ Fab & ~ Fba	3, 4 &I
8	│ │ │ (∃z)(Faz & ~ Fza)	7 ∃I
9	│ │ │ Gaa	6, 8 ⊃E
10	│ │ │ (∃x)Gxx	9 ∃I
11	│ │ │ ~ (∃x)Gxx	2 R
12	│ │ Fba	4–11 ~ E
13	│ Fab ⊃ Fba	3–12 ⊃I
14	│ (∀z)(Faz ⊃ Fza)	13 ∀I

g. Derive: $(\forall x) \sim Fx$

1	$(\forall x)(Fx \supset (\exists y)Gxy)$	Assumption
2	$(\forall x)(\forall y) \sim Gxy$	Assumption
3	Fa	A / ~ I
4	$Fa \supset (\exists y)Gay$	1 ∀E
5	$(\exists y)Gay$	3, 4 ⊃E
6	Gab	A / ∃E
7	$(\forall x)(\forall y) \sim Gxy$	A / ~ I
8	$(\forall y) \sim Gay$	2 ∀E
9	$\sim Gab$	8 ∀E
10	Gab	6 R
11	$\sim (\forall x)(\forall y) \sim Gxy$	7–11 ~ I
12	$\sim (\forall x)(\forall y) \sim Gxy$	5, 6–11 ∃E
13	$(\forall x)(\forall y) \sim Gxy$	2 R
14	$\sim Fa$	3–14 ~I
15	$(\forall x)\sim Fx$	14 ∀I

7. Theorems

a. Derive: $(\forall x)(\exists z)(Fxz \supset Fzx)$

1	Faa	A / ⊃I
2	Faa	1 R
3	$Faa \supset Faa$	1–2 ⊃I
4	$(\exists z)(Faz \supset Fza)$	3 ∃I
5	$(\forall x)(\exists z)(Fxz \supset Fzx)$	4 ∀I

c. Derive: $(\forall x)(\forall y)Gxy \supset (\forall z)Gzz$

1	$(\forall x)(\forall y)Gxy$	A / ⊃I
2	$(\forall y)Gay$	1 ∀E
3	Gaa	2 ∀E
4	$(\forall z)Gzz$	3 ∀I
5	$(\forall x)(\forall y)Gxy \supset (\forall z)Gzz$	1–4 ⊃I

e. Derive: $(\forall x)Lxx \supset (\exists x)(\exists y)(Lxy \,\&\, Lyx)$

1	$(\forall x)Lxx$	A / ⊃I
2	Laa	1 ∀E
3	$Laa \,\&\, Laa$	2, 2 &I
4	$(\exists y)(Lay \,\&\, Lya)$	3 ∃I
5	$(\exists x)(\exists y)(Lxy \,\&\, Lyx)$	4 ∃I
6	$(\forall x)Lxx \supset (\exists x)(\exists y)(Lxy \,\&\, Lyx)$	1–5 ⊃I

#h. The theorem to be proved, '(∃x) (∀y)Fxy ⊃ (∃x) (∃y)Fxy' is a truth-functional compound whose main connective is a material conditional. Therefore, we will use Conditional Introduction as our primary strategy:

Derive: (∃x) (∀y)Fxy ⊃ (∃x) (∃y)Fxy

1	(∃x) (∀y)Fxy	Assumption
G	(∃x) (∃y)Fxy	
G	(∃x) (∀y)Fxy ⊃ (∃x) (∃y)Fxy	1–__ ⊃I

Our current goal is an existentially quantified sentence, '(∃x) (∃y)Fxy'. The most obvious way to obtain it is by two uses of Existential Introduction. Since the sentence on line 1 is an existentially quantified sentence it seems likely we will also be using Existential Elimination. And we know that when we do so, by assuming a substitution instance of '(∃x) (∀y)Fxy', we will have to continue working within that subderivation until we obtain a sentence that does not contain the instantiating constant. This suggests that our current goal, '(∃x) (∀y)Fxy', should also be the goal of our Existential Elimination subderivation, since it contains no constants:

Derive: (∃x) (∀y)Fxy ⊃ (∃x) (∃y)Fxy

1	(∃x) (∀y)Fxy	Assumption
2	(∀y)Fay	A / ∃E
G	(∃x) (∃y)Fxy	
G	(∃x) (∃y)Fxy	2, 3–__ ∃E
G	(∃x) (∀y)Fxy ⊃ (∃x) (∃y)Fxy	1–__ ⊃I

Completing this derivation is now straightforward. We use Universal Elimination on line 2 to produce 'Fab' and then use Existential Introduction twice to produce '(∃x) (∃y)Fxy'.

Derive: (∃x)(∀y)Fxy ⊃ (∃x)(∃y)Fxy

1	(∃x)(∀y)Fxy	Assumption
2	(∀y)Fay	A / ∃E
3	Fab	2 ∀E
4	(∃y)Fay	3 ∃I
5	(∃x)(∃y)Fxy	4 ∃I
6	(∃x)(∃y)Fxy	1, 2–5 ∃E
7	(∃x)(∀y)Fxy ⊃ (∃x)(∃y)Fxy	1–6 ⊃I

Here we do meet all the restrictions on Existential Elimination. The instantiating constant, which is here ‘a’, does not, at the point we use Existential Elimination (line 6) occur in any open assumption. The constant ‘a’ also does not occur in the existentially quantified sentence to which we are applying Existential Elimination, and it does not occur in the sentence derived by Existential Elimination (the sentence on line 6).

It is worth noting that since there are no restrictions on Existential Introduction, we could have entered, at line 3, ‘Faa’ rather than ‘Fab’ (there are also no restrictions on Universal Elimination), and then twice applied Existential Introduction.

i. Derive: (∃x)(∃y)(Lxy ≡ Lyx)

1	Laa	A / ≡I
2	Laa	1 R
3	Laa ≡ Laa	1–2, 1–2 ≡I
4	(∃y)(Lay ≡ Lya)	3 ∃I
5	(∃x)(∃y)(Lxy ≡ Lyx)	4 ∃I

k. Derive: (∀x)(∀y)(∀z)Gxyz ⊃ (∀x)(∀y)(∀z)(Gxyz ⊃ Gzyx)

1	(∀x)(∀y)(∀z)Gxyz	A/ ⊃I
2	Gabc	A / ⊃I
3	(∀y)(∀z)Gcyz	1 ∀E
4	(∀z)Gcbz	3 ∀E
5	Gcba	4 ∀E
6	Gabc ⊃ Gcba	2–5 ⊃I
7	(∀z)(Gabz ⊃ Gzba)	6 ∀I
8	(∀y)(∀z)(Gayz ⊃ Gzya)	7 ∀I
9	(∀x)(∀y)(∀z)(Gxyz ⊃ Gzyz)	
10	(∀x)(∀y)(∀z)Gxyz ⊃ (∀x)(∀y)(∀z)(Gxyz ⊃ Gzyx)	1–9 ⊃I

m. Derive: (∀x)(∀y)(Fxy ≡ Fyx) ⊃ ~ (∃x)(∃y)(Fxy & ~ Fyx)

1	(∀x)(∀y)(Fxy ≡ Fyx)	A / ⊃I
2	(∃x)(∃y)(Fxy & ~ Fyx)	A / ~ I
3	(∃y)(Fay & ~ Fya)	A / ∃E
4	Fab & ~ Fba	A / ∃E
5	(∀x)(∀y)(Fxy ≡ Fyx)	A / ~ I
6	(∀y)(Fay ≡ Fya)	1 ∀E
7	Fab ≡ Fba	6 ∀E
8	Fab	4 &E
9	Fba	7, 8 ≡E
10	~ Fba	4 &E
11	~ (∀x)(∀y)(Fxy ≡ Fyx)	5–10 ~ I
12	~ (∀x)(∀y)(Fxy ≡ Fyx)	3, 4–11 ∃E
13	~ (∀x)(∀y)(Fxy ≡ Fyx)	2, 3–12 ∃E
14	(∀x)(∀y)(Fxy ≡ Fyx)	1 R
15	~ (∃x)(∃y(Fxy & ~ Fyx)	2–14 ~ I
16	(∀x)(∀y)(Fxy ≡ Fyx) ⊃ ~ (∃x)(∃y)(Fxy & ~ Fyx)	1–15 ⊃I

8. Equivalence

a. Derive: (∀x)(Fx ⊃ (∃y)Gya)

1	(∃x)Fx ⊃ (∃y)Gya	Assumption
2	Fa	A / ⊃I
3	(∃x)Fx	2 ∃I
4	(∃y)Gya	1, 3 ⊃E
5	Fa ⊃ (∃y)Gya	2–4 ⊃I
6	(∀x)(Fx ⊃ (∃y)Gya)	5 ∀I

Derive: (∃x)Fx ⊃ (∃y)Gya

1	(∀x)(Fx ⊃ (∃y)Gya)	Assumption
2	(∃x)Fx	A / ⊃I
3	Fb	A / ∃E
4	Fb ⊃ (∃y)Gya	1 ∀E
5	(∃y)Gya	3, 4 ⊃E
6	(∃y)Gya	2, 3–5 ∃E
7	(∃x)Fx ⊃ (∃y)Gya	2–6 ⊃I

#c. To establish that '(∃x)[Fx ⊃ (∀y)Hxy]' and '(∃x)(∀y)(Fx ⊃ Hxy)' are equivalent in *PD* we have to derive each from the unit set of the other. We begin by deriving '(∃x)(∀y)(Fx ⊃ Hxy)' from {(∃x)[Fx ⊃ (∀y)Hxy]}. Since our one primary assumption will be an existentially quantified sentence we will use

Existential Elimination as our primary strategy and do virtually all of the derivation within that strategy:

```
Derive: (∃x)(∀y)(Fx ⊃ Hxy)

1 | (∃x)[Fx ⊃ (∀y)Hxy]                    Assumption
  |------------------
2 |  | Fa ⊃ (∀y)Hay                       A / ∃E
  |  |---------------
  |  |
  |  |
  |  |
G |  | (∃x)(∀y)(Fx ⊃ Hxy)                 I
G | (∃x)(∀y)(Fx ⊃ Hxy)                    1, 2–__ ∃E
```

Our current goal is an existentially quantified sentence. We will try to obtain it by Existential Introduction, and will try to obtain the required substitution instance, which will be a universally quantified sentence, by Universal Introduction:

```
Derive: (∃x)(∀y)(Fx ⊃ Hxy)

1 | (∃x)[Fx ⊃ (∀y)Hxy]                    Assumption
  |------------------
2 |  | Fa ⊃ (∀y)Hay                       A / ∃E
  |  |---------------
3 |  |
  |  |
  |  |
G |  | Fa ⊃ Hab
G |  | (∀y)(Fa ⊃ Hay)                     __ ∀I
G |  | (∃x)(∀y)(Fx ⊃ Hxy)                 __ ∃I
G | (∃x)(∀y)(Fx ⊃ Hxy)                    1, 2–__ ∃E
```

Our goal is now a material conditional, and we can obtain it by using Conditional Introduction and within that strategy Universal Elimination. The completed derivation is

Derive: (∃x) (∀y) (Fx ⊃ Hxy)

1	│ (∃x) [Fx ⊃ (∀y)Hxy]	Assumption
2	│ │ Fa ⊃ (∀y)Hay	A / ∃E
3	│ │ │ Fa	A / ⊃I
4	│ │ │ (∀y)Hay	2, 3 ⊃E
5	│ │ │ Hab	4 ∀E
6	│ │ Fa ⊃ Hab	3–5 ⊃I
7	│ │ (∀y) (Fa ⊃ Hay)	6 ∀I
8	│ │ (∃x) (∀y) (Fx ⊃ Hxy)	7 ∃I
9	│ (∃x) (∀y) (Fx ⊃ Hxy)	1, 2–8 ∃E

At line 5 we used Universal Elimination and in doing so were careful to pick an instantiating constant other than ‘a’ as our instantiating constant. Had we used ‘a’ we would not have been able to do Universal Introduction at line 7 because ‘a’ occurs in an assumption (the one on line 2) that is open as of line 7 and also occurs in line 7 itself.

When we apply Existential Elimination, at line 9, the instantiating constant, which is ‘a,’ does not occur in any open assumption, does not occur in the sentence we obtain at line 9, and of course does not occur in the existentially quantified sentence from which we are working (the sentence on line 1). So all three restrictions on Existential Elimination have been met. Note also that our use of Universal Introduction at line 7 meets both restrictions on that rule. The instantiating constant is ‘b’ and ‘b’ does not occur in any open assumption and does not occur in the sentence we obtain by Universal Introduction, ‘(∀y) (Fa ⊃ Hay)’

The derivation of ‘(∃x) [Fx ⊃ (∀y)Hxy]’ from {(∃x) (∀y) (Fx ⊃ Hxy)} is equally straightforward:

Derive: (∃x) [Fx ⊃ (∀y)Hxy]

1	│ (∃x) (∀y) (Fx ⊃ Hxy)	Assumption
2	│ │ (∀y) (Fa ⊃ Hay)	A / ∃E
3	│ │ │ Fa	A / ⊃I
4	│ │ │ Fa ⊃ Hab	2 ∀E
5	│ │ │ Hab	3, 4 ⊃E
6	│ │ │ (∀y)Hay	5 ∀I
7	│ │ Fa ⊃ (∀y)Hay	3–6 ⊃I
8	│ │ (∃x) [Fx ⊃ (∀y)Hxy]	7 ∃I
9	│ (∃x) [Fx ⊃ (∀y)Hxy]	1, 2–8 ∃E

We have again used Existential Elimination as our primary strategy and have again done the bulk of the work of the derivation within that strategy. We were again careful to pick an instantiating constant other than ‘a’ in doing Universal Elimination at line 4, again because using ‘a’ would prevent us from doing Universal Introduction at line 6.

e. Derive: (∀x)(∀y)(Fxy ≡ ~ Gyx)

1	│ (∀x)(∀y) ~ (Fxy ≡ Gyx)	Assumption
2	│ (∀y) ~ (Fay ≡ Gya)	1 ∀E
3	│ ~ (Fab ≡ Gba)	2 ∀E
4	│ │ Fab	A / ≡I
5	│ │ │ Gba	A / ~ I
6	│ │ │ │ Fab	A / ≡I
7	│ │ │ │ Gab	5 R
8	│ │ │ │ Gab	A / ≡I
9	│ │ │ │ Fab	4 R
10	│ │ │ Fab ≡ Gab	6–7, 8–9 ≡I
11	│ │ │ ~ (Fab ≡ Gab)	3 R
12	│ │ ~ Gba	5–11 ~ I
13	│ │ ~ Gba	A / ≡I
14	│ │ │ ~ Fab	A / ~ E
15	│ │ │ │ Fab	A / ≡I
16	│ │ │ │ │ ~ Gba	A / ~ I
17	│ │ │ │ │ Fba	15 R
18	│ │ │ │ │ ~ Fba	14 R
19	│ │ │ │ Gba	16–18 ~ E
20	│ │ │ │ Gba	A / ≡I
21	│ │ │ │ │ ~ Fba	A / ~ E
22	│ │ │ │ │ Gba	20 R
23	│ │ │ │ │ ~ Gba	13 R
24	│ │ │ │ Fab	21–23 ~ E
25	│ │ │ Fab ≡ Gba	4–12, 13–24 ≡I
26	│ │ │ ~ (Fab ≡ Gba)	3 R
27	│ │ Fab	14–26 ~ E
28	│ Fab ≡ ~ Gba	4–12, 13–27 ≡I
29	│ (∀y)(Fay ≡ ~ Gya)	28 ∀I
30	│ (∀x)(∀y)(Fxy ≡ ~ Gyx)	29 ∀I

Derive: (∀x)(∀y) ~ (Fxy ≡ Gyx)

1	(∀x)(∀y)(Fxy ≡ ~ Gyx)	Assumption
2	Fab ≡ Gba	A / ~ I
3	(∀y)(Fay ≡ ~ Gya)	1 ∀E
4	Fab ≡ ~ Gba	3 ∀E
5	Fab	A ≡I
6	~ Gba	4, 5 ≡E
7	Gba	2, 5 ≡E
8	~ Fab	5–7 ~ I
9	~ Gba	A / ~ E
10	Fab	4, 9 ≡E
11	Gba	2, 10 ≡E
12	~ Gba	9 R
13	Gba	9–12 ~ E
14	Fab	2, 13 ≡E
15	~ (Fab ≡ Gba)	2–14 ~ I
16	(∀y) ~ (Fay ≡ Gya)	15 ∀I
17	(∀x)(∀y) ~ (Fxy ≡ Gyx)	16 ∀I

9. Inconsistency

a. Derive: Tab, ~ Tab
b.

1	(∀x)(∀y)[(Ex & Ey) ⊃ Txy]	Assumption
2	(Ea & Eb) & ~ Tab	Assumption
3	(∀y)[(Ea & Ey) ⊃ Tay]	1 ∀E
4	(Ea & Eb) ⊃ Tab	3 ∀E
5	Ea & Eb	2 & E
6	Tab	4, 5 ⊃E
7	~ Tab	2 &E

c. Derive: (∃x)Fxx, ~ (∃x)Fxx

1	~ (∃x)Fxx	Assumption
2	(∃x)(∀y)Fxy	Assumption
3	(∀y)Fay	A / ∃E
4	Faa	3 ∀E
5	(∃x)Fxx	4 ∃I
6	(∃x)Fxx	2, 3–5 ∃E
7	~ (∃x)Fxx	1 R

e. Derive: $(\forall y) \sim Lay$, $\sim (\forall y) \sim Lay$

1	│ $(\forall x)(\exists y)Lxy$	Assumption
2	│ $(\forall y) \sim Lay$	Assumption
3	│ $(\exists y)Lay$	1 ∀E
4	│ │ Lab	A / ∃E
5	│ │ │ $(\forall y) \sim Lay$	A / ~ I
6	│ │ │ $\sim Lab$	6 ∀E
7	│ │ │ Lab	4 R
8	│ │ $\sim (\forall y) \sim Lay$	5–7 ~ I
9	│ $\sim (\forall y) \sim Lay$	3, 4–8 ∃E
10	│ $(\forall y) \sim Lay$	2 R

g. Derive: $(\exists x) \sim (\exists y)Lyx$, $\sim (\exists x) \sim (\exists y)Lyx$

1	│ $(\forall x)[Hx \supset (\exists y)Lyx]$	Assumption
2	│ $(\exists x) \sim (\exists y)Lyx$	Assumption
3	│ $(\forall x)Hx$	Assumption
4	│ │ $\sim (\exists y)Lya$	A / ∃E
5	│ │ │ $(\exists x) \sim (\exists y)Lyx$	A / ~ I
5	│ │ │ $Ha \supset (\exists y)Lya$	1 ∀E
6	│ │ │ Ha	3 ∀E
7	│ │ │ $(\exists y)Lya$	5, 6 ⊃E
8	│ │ │ $\sim (\exists y)Lya$	4 R
9	│ │ $\sim (\exists x) \sim (\exists y)Lyx$	5–8 ~ I
10	│ $\sim (\exists x) \sim (\exists y)Lyx$	2, 4–9 ∃E
11	│ $(\exists x) \sim (\exists y)Lyx$	2 R

#i. We will now show that the set $\{(\forall x)(\exists y)Fxy, (\exists z) \sim (\exists w)Fzw\}$ is inconsistent in *PD*. This is an interesting problem in several respects. Neither set member is a negation. So it is not obvious which pair of contradictory sentences (the **Q** and ~ **Q** we must derive to show the set is contradictory) we should take as our goal. One of the set members is an existentially quantified sentence, so it is plausible that our derivation will involve an Existential Elimination as its main strategy, with a substitution instance of '$(\exists z) \sim (\exists w)Fzw$' as the assumption of a subderivation. Remembering that it is often useful to do as much of the work of a derivation as possible within an Existential Elimination subderivation we will make Existential Elimination our primary strategy:

Derive: ?, ?

1	│ $(\forall x)(\exists y)Fxy$	Assumption
2	│ $(\exists z) \sim (\exists w)Fzw$	Assumption
3	│ │ $\sim (\exists w)Faw$	A / ∃E

Our new assumption is a negation, but that is obviously no hope of moving that sentence out from within the scope of our subderivation so that it can play the role of ~ **Q** in our derivation – no hope because it obviously contains the instantiating constant ‘a’. A better strategy is to try to obtain a negation within the scope of the Existential Elimination strategy that does not contain the constant ‘a’. The obviously useful negation is ‘~ (∀x)(∃y)Fxy’ because we can obtain the sentence of which it is the negation, ‘(∀x)(∃y)Fxy’ by Reiteration on line 1. So we will proceed as follows:

```
Derive: (∀x)(∃y)Fxy, ~ (∀x)(∃y)Fxy

1 | (∀x)(∃y)Fxy                    Assumption
2 | (∃z) ~ (∃w)Fzw                 Assumption
  |-------------
3 | | ~ (∃w)Faw                    A / ∃E
  | |-----------
4 | | | (∀x)(∃y)Fxy                A / ~ I
  | | |-----------
  | | |
  | | |
  | | |
G | | ~ (∀x)(∃y)Fxy                ___–___ ~ I
G | ~ (∀x)(∃y)Fxy                  2, 3–___ ∃E
  | (∀x)(∃y)Fxy                    1 R
```

We now need to derive a sentence and its negation within the scope of the assumption on line 4. There is no reason not to use the negation on line 3. We will do so, making our new goal ‘(∃w)Faw’:

```
Derive: (∀x)(∃y)Fxy, ~ (∀x)(∃y)Fxy

1 | (∀x)(∃y)Fxy                    Assumption
2 | (∃z) ~ (∃w)Fzw                 Assumption
  |-------------
3 | | ~ (∃w)Faw                    A / ∃E
  | |-----------
4 | | | (∀x)(∃y)Fxy                A / ~ I
  | | |-----------
  | | |
  | | |
  | | |
G | | | (∃w)Faw
  | | | ~ (∃w)Faw                  3 R
G | | ~ (∀x)(∃y)Fxy                ___–___ ~ I
G | ~ (∀x)(∃y)Fxy                  2, 3–___ ∃E
  | (∀x)(∃y)Fxy                    1 R
```

From line 1 we can obtain '(∃y)Fay' by Universal Elimination. And we can move from '(∃y)Fay' to '(∃w)Faw' by an Existential Elimination strategy. Our completed derivation is

Derive: (∀x)(∃y)Fxy, ~ (∀x)(∃y)Fxy

1	│ (∀x)(∃y)Fxy	Assumption
2	│ (∃z) ~ (∃w)Fzw	Assumption
3	│ │ ~ (∃w)Faw	A / ∃E
4	│ │ │ (∀x)(∃y)Fxy	A / ~ I
5	│ │ │ (∃y)Fay	1 ∀E
6	│ │ │ │ Fab	A / ∃E
7	│ │ │ │ (∃w)Faw	6 ∃I
8	│ │ │ (∃w)Faw	5, 6–7 ∃E
9	│ │ │ ~ (∃w)Faw	3 R
10	│ │ ~ (∀x)(∃y)Fxy	4–9 ~ I
11	│ ~ (∀x)(∃y)Fxy	2, 3–10 ∃E
12	│ (∀x)(∃y)Fxy	1 R

We have used Existential Elimination twice and in both instances we met all restrictions on that rule. In the first use, at line 8, the instantiating constant is 'b' and 'b' does not occur in either line 5 or line 8 and it does not, as of line 8, occur in any open assumption.

k. Derive: (∀x)(∀y)(Fxy ∨ Gxy), ~ (∀x)(∀y)(Fxy ∨ Gxy)

1	│ (∀x)(∀y)(Fxy ∨ Gxy)	Assumption
2	│ (∃x)(∃y)(~ Fxy & ~ Gxy)	Assumption
3	│ │ (∃y)(~ Fay & ~ Gay)	A / ∃E
4	│ │ │ ~ Fab & ~ Gab	A / ∃E
5	│ │ │ (∀y)(Fay ∨ Gay)	1 ∀E
6	│ │ │ Fab ∨ Gab	5 ∀E
7	│ │ │ │ Fab	A / ∨E
8	│ │ │ │ │ (∀x)(∀y)(Fxy ∨ Gxy)	A / ~ I
9	│ │ │ │ │ Fab	7 R
10	│ │ │ │ │ ~ Fab	4 &E
11	│ │ │ │ ~ (∀x)(∀y)(Fxy ∨ Gxy)	8–10 ~ I
12	│ │ │ │ Gab	A ∨E
13	│ │ │ │ │ (∀x)(∀y)(Fxy ∨ Gxy)	A / ~ I
14	│ │ │ │ │ Gab	14 R
15	│ │ │ │ │ ~ Gab	4 &E
16	│ │ │ │ ~ (∀x)(∀y)(Fxy ∨ Gxy)	13–15 ~ I
17	│ │ │ ~ (∀x)(∀y)(Fxy ∨ Gxy)	6, 7–11, 12–16 ∨E
18	│ │ ~ (∀x)(∀y)(Fxy ∨ Gxy)	3, 4–17 ∃E
19	│ ~ (∀x)(∀y)(Fxy ∨ Gxy)	2, 3–18 ∃E
20	│ (∀x)(∀y)(Fxy ∨ Gxy)	1 R

10.3E

1. Derivability

a. Derive: (∃y)(~ Fy ∨ ~ Gy)

1	~ (∀y)(Fy & Gy)	Assumption
2	(∃y) ~ (Fy & Gy)	1 QN
3	(∃y)(~ Fy ∨ ~ Gy)	2 DeM

c. Derive: (∃z)(Az & ~ Cz)

1	(∃z)(Gz & Az)	Assumption
2	(∀y)(Cy ⊃ ~ Gy)	Assumption
3	Gh & Ah	A / ∃E
4	Ch ⊃ ~ Gh	2 ∀E
5	Gh	3 &E
6	~ ~ Gh	5 DN
7	~ Ch	4, 6 MT
8	Ah	3 &E
9	Ah & ~ Ch	8, 7 &I
10	(∃z)(Az & ~ Cz)	9 ∃I
11	(∃z)(Az & ~ Cz)	1, 3–10 ∃E

e. Derive: (∃x)Cxb

1	(∀x)[(~ Cxb ∨ Hx) ⊃ Lxx]	Assumption
2	(∃y) ~ Lyy	Assumption
3	~ Lmm	A / ∃E
4	(~ Cmb ∨ Hm) ⊃ Lmm	1 ∀E
5	~ (~ Cmb ∨ Hm)	3, 4 MT
6	~ ~ Cmb & ~ Hm	5 DeM
7	~ ~ Cmb	6 &E
8	Cmb	7 DN
9	(∃x)Cxb	8 ∃I
10	(∃x)Cxb	2, 3–9 ∃E

2. Validity

a. Derive: (∀y) ~ (Hby ∨ Ryy)

1	(∀y) ~ Jx	Assumption
2	(∃y)(Hby ∨ Ryy) ⊃ (∃x)Jx	Assumption
3	~ (∃x)Jx	1 QN
4	~ (∃y)(Hby ∨ Ryy)	2, 3 MT
5	(∀y) ~ (Hby ∨ Ryy)	4 QN

c. Derive: (∀x)(∀y)Hxy & (∀x) ~ Tx

1	(∀x) ~ ((∀y)Hyx ∨ Tx)	Assumption
2	~ (∃y)(Ty ∨ (∃x) ~ Hxy)	Assumption
3	(∀y) ~ (Ty ∨ (∃x) ~ Hxy)	2 QN
4	~ (Ta ∨ (∃x) ~ Hxa)	3 ∀E
5	~ Ta & ~ (∃x) ~ Hxa	4 DeM
6	~ (∃x) ~ Hxa	5 &E
7	(∀x) ~ ~ Hxa	6 QN
8	~ ~ Hba	7 ∀E
9	Hba	8 DN
10	(∀y)Hby	9 ∀I
11	(∀x)(∀y)Hxy	10 ∀I
12	~ Ta	5 &E
13	(∀x) ~ Tx	12 ∀I
14	(∀x)(∀y)Hxy & (∀x) ~ Tx	11, 13 &I

e. Derive: (∃x) ~ Kxx

1	(∀z)[Kzz ⊃ (Mz & Nz)]	Assumption
2	(∃z) ~ Nz	Assumption
3	~ Ng	A / ∃E
4	Kgg ⊃ (Mg & Ng)	1 ∀E
5	~ Mg ∨ ~ Ng	3 ∨I
6	~ (Mg & Ng)	5 DeM
7	~ Kgg	4, 6 MT
8	(∃x) ~ Kxx	7 ∃I
9	(∃x) ~ Kxx	2, 3–8 ∃E

g. Derive: (∃w)(Gw & Bw) ⊃ (∀y)(Lyy ⊃ ~ Ay)

1	(∃z)Gz ⊃ (∀w)(Lww ⊃ ~ Hw)	Assumption
2	(∃x)Bx ⊃ (∀y)(Ay ⊃ Hy)	Assumption
3	(∃w)(Gw & Bw)	A / ⊃I
4	Gm & Bm	A / ∃E
5	Gm	4 &E
6	(∃z)Gz	5 ∃I
7	(∀w)(Lww ⊃ ~ Hw)	1, 6 ⊃E
8	Lcc ⊃ ~ Hc	7 ∀E
9	Bm	4 &E
10	(∃x)Bx	9 ∃I
11	(∀y)(Ay ⊃ Hy)	2, 10 ⊃E
12	Ac ⊃ Hc	11 ∀E
13	~ Hc ⊃ ~ Ac	12 Trans
14	Lcc ⊃ ~ Ac	8, 13 HS
15	(∀y)(Lyy ⊃ ~ Ay)	14 ∀I
16	(∀y)(Lyy ⊃ ~ Ay)	3, 4–15 ∃E
17	(∃w)(Gw & Bw) ⊃ (∀y)(Lyy ⊃ ~ Ay)	3–16 ⊃I

i. Derive: $\sim (\forall x)(\forall y)Bxy \supset (\forall x)(\sim Gx \vee \sim Hx)$

1	$\sim (\forall x)(\sim Gx \vee \sim Hx) \supset (\forall x)[Cx \,\&\, (\forall y)(Ly \supset Axy)]$	Assumption
2	$(\exists x)\ [Hx \,\&\, (\forall y)(Ly \supset Axy)] \supset (\forall x)(Fx \,\&\, (\forall y)Bxy)$	Assumption
3	$\sim (\forall x)(\sim Gx \vee \sim Hx)$	A / ⊃I
4	$(\exists x) \sim (\sim Gx \vee \sim Hx)$	3 QN
5	$\sim (\sim Gi \vee \sim Hi)$	A / ∃I
6	$\sim \sim Gi \,\&\, \sim \sim Hi$	5 DeM
7	$\sim \sim Hi$	6 &E
8	Hi	7 DN
9	$(\forall x)[Cx \,\&\, (\forall y)(Ly \supset Axy)]$	1, 3 ⊃E
10	$Ci \,\&\, (\forall y)(Ly \supset Aiy)$	9 ∀E
11	$(\forall y)(Ly \supset Aiy)$	10 &E
12	$Hi \,\&\, (\forall y)(Ly \supset Aiy)$	8, 11 &I
13	$(\exists x)[Hx \,\&\, (\forall y)(Ly \supset Axy)]$	12 ∃I
14	$(\forall x)(Fx \,\&\, (\forall y)Bxy)$	2, 13 ⊃E
15	$Fj \,\&\, (\forall y)Bjy$	14 ∀E
16	$(\forall y)Bjy$	15 &E
17	$(\forall x)(\forall y)Bxy$	16 ∀I
18	$(\forall x)(\forall y)Bxy$	4, 5–17 ∃E
19	$\sim (\forall x)(\sim Gx \vee \sim Hx) \supset (\forall x)(\forall y)Bxy$	3–18 ⊃I
20	$\sim (\forall x)(\forall y)Bxy \supset \sim \sim (\forall x)(\sim Gx \vee \sim Hx)$	19 Trans
21	$\sim (\forall x)(\forall y)Bxy \supset (\forall x)(\sim Gx \vee \sim Hx)$	20 DN

3. Theorems

a. Derive: $(\forall x)(Ax \supset Bx) \supset (\forall x)(Bx \vee \sim Ax)$

1	$(\forall x)(Ax \supset Bx)$	A / ⊃I
2	$(\forall x)(\sim Ax \vee Bx)$	1 Impl
3	$(\forall x)(Bx \vee \sim Ax)$	2 Com
4	$(\forall x)(Ax \supset Bx) \supset (\forall x)(Bx \vee \sim Ax)$	1–3 ⊃I

c. Derive: $\sim (\exists x)(Ax \vee Bx) \supset (\forall x) \sim Ax$

1	$\sim (\exists x)(Ax \vee Bx)$	A / ⊃I
2	$(\forall x) \sim (Ax \vee Bx)$	1 QN
3	$\sim (Ac \vee Bc)$	2 ∀E
4	$\sim Ac \,\&\, \sim Bc$	3 DeM
5	$\sim Ac$	4 &E
6	$(\forall x) \sim Ax$	5 ∀I
7	$\sim (\exists x)(Ax \vee Bx) \supset (\forall x) \sim Ax$	1–6 ⊃I

e. Derive: $((\exists x)Ax \supset (\exists x)Bx) \supset (\exists x)(Ax \supset Bx)$

1	$\sim (\exists x)(Ax \supset Bx)$	A / $\supset$I
2	$(\forall x) \sim (Ax \supset Bx)$	1 QN
3	$\sim (Ac \supset Bc)$	2 $\forall$E
4	$\sim (\sim Ac \vee Bc)$	3 Impl
5	$\sim \sim Ac \;\&\; \sim Bc$	4 DeM
6	$\sim \sim Ac$	5 &E
7	$(\exists x) \sim \sim Ax$	6 $\exists$I
8	$\sim (\forall x) \sim Ax$	7 QN
9	$\sim \sim (\exists x)Ax$	8 QN
10	$\sim Bc$	5 &E
11	$(\forall x) \sim Bx$	10 $\forall$I
12	$\sim (\exists x)Bx$	11 QN
13	$\sim \sim (\exists x)Ax \;\&\; \sim (\exists x)Bx$	9, 12 &I
14	$\sim (\sim (\exists x)Ax \vee (\exists x)Bx)$	13 DeM
15	$\sim ((\exists x)Ax \supset (\exists x)Bx)$	14 Impl
16	$\sim (\exists x)(Ax \supset Bx) \supset \sim ((\exists x)Ax \supset (\exists x)Bx)$	1–15 $\supset$I
17	$((\exists x)Ax \supset (\exists x)Bx) \supset (\exists x)(Ax \supset Bx)$	16 Trans

4. Equivalence

a. Derive: $(\exists x)(Ax \;\&\; \sim Bx)$

1	$\sim (\forall x)(Ax \supset Bx)$	Assumption
2	$(\exists x) \sim (Ax \supset Bx)$	1 QN
3	$(\exists x) \sim (\sim Ax \vee Bx)$	2 Impl
4	$(\exists x)(\sim \sim Ax \;\&\; \sim Bx)$	3 DeM
5	$(\exists x)(Ax \;\&\; \sim Bx)$	4 DN

Derive: $\sim (\forall x)(Ax \supset Bx)$

1	$(\exists x)(Ax \;\&\; \sim Bx)$	Assumption
2	$(\exists x)(\sim \sim Ax \;\&\; \sim Bx)$	1 DN
3	$(\exists x) \sim (\sim Ax \vee Bx)$	2 DeM
4	$(\exists x) \sim (Ax \supset Bx)$	3 Impl
5	$\sim (\forall x)(Ax \supset Bx)$	4 QN

c. Derive: $(\exists x)[\sim Ax \vee (\sim Cx \supset \sim Bx)]$

1	$\sim (\forall x) \sim [(Ax \;\&\; Bx) \supset Cx]$	Assumption
2	$(\exists x) \sim \sim [(Ax \;\&\; Bx) \supset Cx]$	1 QN
3	$(\exists x)[(Ax \;\&\; Bx) \supset Cx]$	2 DN
4	$(\exists x)[Ax \supset (Bx \supset Cx)]$	3 Exp
5	$(\exists x)[\sim Ax \vee (Bx \supset Cx)]$	4 Impl
6	$(\exists x)[\sim Ax \vee (\sim Cx \supset \sim Bx)]$	5 Trans

Derive: $\sim (\forall x) \sim [(Ax \,\&\, Bx) \supset Cx]$

1	$(\exists x)[\sim Ax \vee (\sim Cx \supset \sim Bx)]$	Assumption
2	$(\exists x)[\sim Ax \vee (Bx \supset Cx)]$	1 Trans
3	$(\exists x)[Ax \supset (Bx \supset Cx)]$	2 Impl
4	$(\exists x)[(Ax \,\&\, Bx) \supset Cx]$	3 Exp
5	$\sim \sim (\exists x)[(Ax \,\&\, Bx) \supset Cx]$	4 DN
6	$\sim (\forall x) \sim [(Ax \,\&\, Bx) \supset Cx]$	5 QN

e. Derive: $\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (Ax \vee Bx)]$

1	$(\forall x)(Ax \equiv Bx)$	Assumption
2	$\sim \sim (\forall x)(Ax \equiv Bx)$	1 DN
3	$\sim (\exists x) \sim (Ax \equiv Bx)$	2 QN
4	$\sim (\exists x) \sim [(Ax \,\&\, Bx) \vee (\sim Ax \,\&\, \sim Bx)]$	3 Equiv
5	$\sim (\exists x)[\sim (Ax \,\&\, Bx) \,\&\, \sim (\sim Ax \,\&\, \sim Bx)]$	4 DeM
6	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, \sim (\sim Ax \,\&\, \sim Bx)]$	5 DeM
7	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (\sim \sim Ax \vee \sim \sim Bx)]$	6 DeM
8	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (Ax \vee \sim \sim Bx)]$	7 DN
9	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (Ax \vee Bx)]$	8 DN

Derive: $(\forall x)(Ax \equiv Bx)$

1	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (Ax \vee Bx)]$	Assumption
2	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (Ax \vee \sim \sim Bx)]$	1 DN
3	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, (\sim \sim Ax \vee \sim \sim Bx)]$	2 DN
4	$\sim (\exists x)[(\sim Ax \vee \sim Bx) \,\&\, \sim (\sim Ax \,\&\, \sim Bx)]$	3 DeM
5	$\sim (\exists x)[\sim (Ax \,\&\, Bx) \,\&\, \sim (\sim Ax \,\&\, \sim Bx)]$	4 DeM
6	$\sim (\exists x) \sim [(Ax \,\&\, Bx) \vee (\sim Ax \,\&\, \sim Bx)]$	5 DeM
7	$\sim (\exists x) \sim (Ax \equiv Bx)$	6 Equiv
8	$\sim \sim (\forall x)(Ax \equiv Bx)$	7 QN
9	$(\forall x)(Ax \equiv Bx)$	8 DN

5. Inconsistency

a. Derive: Jc, $\sim Jc$

1	$[(\forall x)(Mx \equiv Jx) \,\&\, \sim Mc] \,\&\, (\forall x)Jx$	Assumption
2	$(\forall x)(Mx \equiv Jx) \,\&\, \sim Mc$	1 &E
3	$(\forall x)(Mx \equiv Jx)$	2 &E
4	$Mc \equiv Jc$	3 ∀E
5	$(Mc \supset Jc) \,\&\, (Jc \supset Mc)$	4 Equiv
6	$Jc \supset Mc$	5 &E
7	$\sim Mc$	2 &E
8	$\sim Jc$	6, 7 MT
9	$(\forall x)Jx$	1 &E
10	Jc	9 ∀E

c. Derive: $(\exists w)Cww$, $\sim (\exists w)Cww$

1	$(\forall x)(\forall y)Lxy \supset \sim (\exists z)Tz$	Assumption
2	$(\forall x)(\forall y)Lxy \supset ((\exists w)Cww \vee (\exists z)Tz)$	Assumption
3	$(\sim (\forall x)(\forall y)Lxy \vee (\forall z)Bzzk)\ \&$ $(\sim (\forall z)Bzzk \vee \sim (\exists w)Cww)$	Assumption
4	$(\forall x)(\forall y)Lxy$	Assumption
5	$\sim (\exists z)Tz$	1, 4 $\supset$E
6	$(\exists w)Cww \vee (\exists z)Tz$	2, 4 $\supset$E
7	$(\exists w)Cww$	5, 6 DS
8	$\sim (\forall x)(\forall y)Lxy \vee (\forall z)Bzzk$	3 &E
9	$(\forall x)(\forall y)Lxy \supset (\forall z)Bzzk$	8 Impl
10	$(\forall z)Bzzk$	4, 9 $\supset$E
11	$\sim (\forall z)Bzzk \vee \sim (\exists w)Cww$	3 &E
12	$(\forall z)Bzzk \supset \sim (\exists w)Cww$	11 Impl
13	$\sim (\exists w)Cww$	10, 12 $\supset$E

e. Derive: Hc, $\sim Hc$

1	$(\forall x)(\forall y)(Gxy \supset Hc)$	Assumption
2	$(\exists x)Gix\ \&\ (\forall x)(\forall y)(\forall z)Lxyz$	Assumption
3	$\sim Lcib \vee \sim (Hc \vee Hc)$	Assumption
4	$(\exists x)Gix$	2 &E
5	$\quad Gik$	A / $\supset$I
6	$\quad (\forall y)(Giy \supset Hc)$	1 $\forall$E
7	$\quad Gik \supset Hc$	6 $\forall$E
8	$\quad Hc$	5, 7 $\supset$E
9	Hc	4, 5–8 $\exists$E
10	$(\forall x)(\forall y)(\forall z)Lxyz$	2 &E
11	$(\forall y)(\forall z)Lcyz$	10 $\forall$E
12	$(\forall z)Lciz$	11 $\forall$E
13	$Lcib$	12 $\forall$E
14	$\sim \sim Lcib$	13 DN
15	$\sim (Hc \vee Hc)$	3, 14 DS
16	$\sim Hc$	15 Idem

6. a. Suppose there is a sentence on an accessible line **i** of a derivation to which Universal Elimination can be properly applied at line **n**. The sentence that would be derived by Universal Elimination can also be derived by using the routine beginning at line **n**:

i	$(\forall \mathbf{x})\mathbf{P}$	
n	$\quad \sim \mathbf{P}(\mathbf{a}/\mathbf{x})$	A / $\sim$ E
n + 1	$\quad (\exists x) \sim \mathbf{P}$	**n** $\exists$I
n + 2	$\quad \sim (\forall \mathbf{x})\mathbf{P}$	**n** + 1 QN
n + 3	$\quad (\forall \mathbf{x})\mathbf{P}$	**i** R
n + 4	$\mathbf{P}(\mathbf{a}/\mathbf{x})$	**n** − **n** + 3 $\sim$ E

Suppose there is a sentence on an accessible line **i** of a derivation to which Universal Introduction can be properly applied at line **n**. The sentence that would be derived by Universal Introduction can also be derived by using the routine beginning at line **n**:

i	**P**(**a**/**x**)	
n	~ (∀**x**)**P**	A / ~ E
n + 1	(∃**x**) ~ **P**	**n** QN
n + 2	~ **P**(**a**/**x**)	A / ~ E
n + 3	~ (∀**x**)**P**	A / ~ E
n + 4	**P**(**a**/**x**)	**i** R
n + 5	~ **P**(**a**/**x**)	**n** + 2 R
n + 6	(∀**x**)**P**	**n** + 3 − **n** + 5 ~ E
n + 7	(∀**x**)**P**	**n** + 1, **n** + 2 − **n** + 6 ∃E
n + 8	~ (∀**x**)**P**	**n** R
n + 9	(∀**x**)**P**	**n** − **n** + 8 ~ E

No restriction on the use of Existential Elimination was violated at line **n** + 7. We assumed that we could have applied Universal Introduction at line **n** to **P**(**a**/**x**) on line **i**. So **a** does not occur in any undischarged assumption prior to line **n**, and **a** does not occur in (∀**x**)**P**. So **a** does not occur in **P**. Hence

(i) **a** does not occur in any undischarged assumption prior to **n** + 7. Note that the assumptions on lines **n** + 2 and **n** + 3 have been discharged and that **a** cannot occur in the assumption on line **n**, for **a** does not occur in **P**.

(ii) **a** does not occur in (∃**x**) ~ **P**, for **a** does not occur in **P**.

(iii) **a** does not occur in (∀**x**)**P**, for **a** does not occur in **P**.

10.4E Exercises

1. Theorems

a. Derive: a = b ⊃ b = a

1	a = b	Assumption
2	a = a	1, 1 =E
3	b = a	1, 2 =E
4	a = b ⊃ b = a	1–3 ⊃I

c. Derive: (~ a = b & b = c) ⊃ ~ a = c

1	~ a = b & b = c	Assumption
2	~ a = b	1 &E
3	b = c	1 &E
4	~ a = c	2, 3 =E
5	(~ a = b & b = c) ⊃ ~ a = c)	1–4 ⊃I

e. Derive: ~ a = c ⊃ (~ a = b ∨ ~ b = c)

```
 1  | | ~ a = c                                Assumption
    | |------------------------
 2  | | | ~ (~ a = b ∨ ~ b = c)                 A / ~ E
    | | |----------------------
 3  | | | | ~ a = b                             A / ~ E
    | | | |--------------------
 4  | | | | ~ a = b ∨ ~ b = c                   3 ∨I
 5  | | | | ~ (~ a = b ∨ ~ b = c)               3 R
 6  | | | a = b                                 3–5 ~ E
 7  | | | ~ b = c                               1, 6 =E
 8  | | | ~ a = b ∨ ~ b = c                     7 ∨I
 9  | | | ~ (~ a = b ∨ ~ b = c)                 2 R
10  | | ~ a = b ∨ ~ b = c                       2–9 ~ E
11  | ~ a = c ⊃ (~ a = b ∨ ~ b = c)             1–10 ⊃I
```

2. Validity

a. Derive: ~ (∀x)Bxx

```
1  | a = b & ~ Bab                 Assumption
   |--------------
2  | ~ Bab                         1 &E
3  | a = b                         1 &E
4  | | (∀x)Bxx                     A / ~ I
   | |----------
5  | | Baa                         4 ∀E
6  | | ~ Baa                       2, 3 =E
7  | ~ (∀x)Bxx                     4–6 ~ I
```

c. Derive: Hii

```
 1  | (∀z)[Gz ⊃ (∀y)(Ky ⊃ Hzy)]        Assumption
 2  | (Ki & Gj) & i = j                Assumption
    |----------------------------
 3  | Gj ⊃ (∀y)(Ky ⊃ Hjy)              1 ∀E
 4  | Ki & Gj                          2 &E
 5  | Gj                               4 &E
 6  | (∀y)(Ky ⊃ Hjy)                   3, 5 ⊃E
 7  | Ki ⊃ Hji                         7 ∀E
 8  | Ki                               4 &E
 9  | Hji                              7, 8 ⊃E
10  | i = j                            2 &E
11  | Hii                              9, 10 =E
```

e. Derive: $Ka \vee \sim Kb$

1	$a = b$	Assumption
2	$\sim (Ka \vee \sim Ka)$	A / ~ E
3	Ka	A / ~ I
4	$Ka \vee \sim Ka$	3 ∨I
5	$\sim (Ka \vee \sim Ka)$	2 R
6	$\sim Ka$	3–5 ~ I
7	$Ka \vee \sim Ka$	6 ∨I
8	$\sim (Ka \vee \sim Ka)$	2 R
9	$Ka \vee \sim Ka$	2–8 ~ E
10	$Ka \vee \sim Kb$	1, 9 =E

3. Theorems

a. Derive: $(\forall x)(x = x \vee \sim x = x)$

1	$(\forall x)x = x$	=I
2	$a = a$	1 ∀E
3	$a = a \vee \sim a = a$	2 ∨I
4	$(\forall x)(x = x \vee \sim x = x)$	3 ∀I

c. Derive: $(\forall x)(\forall y)(x = y \equiv y = x)$

1	$a = b$	A / ≡I
2	$a = a$	1, 1 =E
3	$b = a$	1, 2 =E
4	$b = a$	A / ≡I
5	$b = b$	4, 4 =E
6	$a = b$	4, 5 =E
7	$a = b \equiv b = a$	1–3, 4–6 ≡I
8	$(\forall y)(a = y \equiv y = a)$	7 ∀I
9	$(\forall x)(\forall y)(x = y \equiv y = x)$	8 ∀I

e. Derive: $\sim (\exists x) \sim x = x$

1	$(\exists x) \sim x = x$	A / ~ I
2	$\sim a = a$	A / ∃E
3	$(\exists x) \sim x = x$	A / ~ I
4	$(\forall x)x = x$	=I
5	$a = a$	4 ∀E
6	$\sim a = a$	2 R
7	$\sim (\exists x) \sim x = x$	3–6, ~ I
8	$\sim (\exists x) \sim x = x$	1, 2–7 ∃E
9	$(\exists x) \sim x = x$	1 R
10	$\sim (\exists x) \sim x = x$	1–9 ~ I

4. Validity

a. Derive: (∃x)(∃y)[(Ex & Ey) & ~ x = y]

1	~ t = f	Assumption
2	Et & Ef	Assumption
3	(Et & Ef) & ~ t = f	1, 2 &I
4	(∃y)[(Et & Ey) & ~ t = y]	3 ∃I
5	(∃x)(∃y)[(Ex & Ey) & ~ x = y]	4 ∃I

c. Derive: ~ s = b

1	~ Ass & Aqb	Assumption
2	(∀x)[(∃y)Ayx ⊃ Abx]	Assumption
3	s = b	A / ~ I
4	(∃y)Ayb ⊃ Abb	2 ∀E
5	Aqb	1 &E
6	(∃y)Ayb	5 ∃I
7	Abb	4, 6 ⊃E
8	~ Ass	1 &E
9	~ Abb	3, 8 =E
10	~ s = b	3–9 ~ I

e. Derive: (∃x)[(Rxe & Pxa) & (~ x = e & ~ x = a)]

1	(∃x)(Rxe & Pxa)	Assumption
2	~ Ree	Assumption
3	~ Paa	Assumption
4	Rie & Pia	A / ∃E
5	i = e	A / ~ I
6	Rie	4 &E
7	Ree	5, 6 =E
8	~ Ree	2 R
9	~ i = e	5–8 ~ I
10	i = a	A / ~ I
11	Pia	4 &E
12	Paa	10, 11 =E
13	~ Paa	3 R
14	~ i = a	10–13 ~ I
15	~ i = e & ~ i = a	9, 14 &I
16	(Rie & Pia) & (~ i = e & ~ i = a)	4, 15 &I
17	(∃x)[(Rxe & Pxa) & (~ x = e & ~ x = a)]	16 ∃I
18	(∃x)[(Rxe & Pxa) & (~ x = e & ~ x = a)]	1, 4–17 ∃E

5.a.

1	$(\exists x)Sx$	Assumption
2	$Sg(f)$	A / ∃E
3	$(\exists x)Sg(x)$	2 ∃I
4	$(\exists x)Sg(x)$	1, 2–3 ∃E

Line 2 is a mistake as an instantiating individual constant must be used, *not* a closed complex term.

c. Correctly done.

e.

1	$(\forall x)Lxxx$	Assumption
2	$Lf(a,a)a$	1 ∀E
3	$(\forall x)Lf(x,x)x$	2 ∀I

Line 2 is a mistake. Universal Elimination does not permit using both a closed complex term and at the same time an individual constant in the substitution instance, not to mention that all three occurrences of the variable ‘x’ must be replaced.

g.

1	$(\forall x)Rf(x,x)$	Assumption
2	$Rf(c,c)$	1 ∀E
3	$(\forall y)Ry$	2 ∀I

Line 3 is a mistake. Universal Introduction cannot be applied using a closed complex term.

i. Correctly done.

6. Theorems in *PDE*:

a. Derive: $(\forall x)(\exists y)f(x) = y$

1	$(\forall x)x = x$	=I
2	$f(a) = f(a)$	1 ∀E
3	$(\exists y)f(a) = y$	2 ∃I
4	$(\forall x)(\exists y)f(x) = y$	3 ∀I

c. Derive: $(\forall x)Ff(x) \supset (\forall x)Ff(g(x))$

1	$(\forall x)Ff(x)$	A / ⊃I
2	$Ff(g(a))$	1 ∀E
3	$(\forall x)Ff(g(x))$	2 ∀I
4	$(\forall x)Ff(x) \supset (\forall x)Ff(g(x))$	1–3 ⊃I

e. Derive: $(\forall x)(f(f(x)) = x \supset f(f(f(f(x)))) = x)$

1	$f(f(a)) = a$	A / $\supset$I
2	$f(f(f(f(a)))) = a$	1, 1 =E
3	$f(f(a)) = a \supset f(f(f(f(a)))) = a$	1–2 $\supset$I
4	$(\forall x)(f(f(x)) = x \supset f(f(f(f(x)))) = x)$	3 $\forall$I

g. Derive: $(\forall x)(\forall y)[(f(x) = y \,\&\, f(y) = x) \supset x = f(f(x))]$

1	$f(a) = b \,\&\, f(b) = a$	A / $\supset$ I
2	$f(b) = a$	1 &E
3	$f(b) = f(b)$	2, 2 =E
4	$a = f(b)$	2, 3 =E
5	$f(a) = b$	1 &E
6	$a = f(f(a))$	4, 5 =E
7	$(f(a) = b \,\&\, f(b) = a) \supset a = f(f(a))$	1–6 $\supset$ I
8	$(\forall y)[(f(a) = y \,\&\, f(y) = a) \supset a = f(f(a))]$	7 $\forall$I
9	$(\forall x)(\forall y)[(f(x) = y \,\&\, f(y) = x) \supset x = f(f(x))]$	8 $\forall$I

7. Validity in *PDE*:

a. Derive: $(\forall x)Gf(x)f(f(x))$

1	$(\forall x)(Bx \supset Gxf(x))$	Assumption
2	$(\forall x)Bf(x)$	Assumption
3	$Bf(a) \supset Gf(a)f(f(a))$	1 $\forall$E
4	$Bf(a)$	2 $\forall$I
5	$Gf(a)f(f(a))$	3, 4 $\supset$E
6	$(\forall x)Gf(x)f(f(x))$	5 $\forall$I

c. Derive: $\sim f(a) = b$

1	$(\forall x)(\forall y)(f(x) = y \supset Myxc)$	Assumption
2	$\sim Mbac \,\&\, \sim Mabc$	Assumption
3	$(\forall y)(f(a) = y \supset Myac)$	1 $\forall$E
4	$f(a) = b \supset Mbac$	3 $\forall$E
5	$f(a) = b$	A / $\sim$ I
6	$Mbac$	4, 5 $\supset$E
7	$\sim Mbac$	2 &E
8	$\sim f(a) = b$	5–7 $\sim$ I

e. Derive: $(\exists x)Lxf(x)g(x)$

1	$(\exists x)(\forall y)(\forall z)Lxyz$	Assumption
2	$(\forall y)(\forall z)Layz$	A / ∃E
3	$(\forall z)Laf(a)z$	2 ∀E
4	$Laf(a)g(a)$	3 ∀E
5	$(\exists x)Lxf(x)g(x)$	4 ∃I
6	$(\exists x)Lxf(x)g(x)$	1, 2–5 ∃E

g. Derive: $(\forall x)Df(x)f(x)$

1	$(\forall x)[Zx \supset (\forall y)(\sim Dxy \equiv Hf(f(y)))]$	Assumption
2	$(\forall x)(Zx \;\&\; \sim Hx)$	Assumption
3	$Zf(a) \supset (\forall y)(\sim Df(a)y \equiv Hf(f(y)))$	1 ∀E
4	$Zf(a) \;\&\; \sim Hf(a)$	2 ∀E
5	$Zf(a)$	4 &E
6	$(\forall y)(\sim Df(a)y \equiv Hf(f(y)))$	3, 5 ⊃E
7	$\sim Df(a)f(a) \equiv Hf(f(f(a)))$	6 ∀E
8	$\sim Df(a)f(a)$	A / ~ E
9	$Hf(f(f(a)))$	7, 8 ≡E
10	$Zf(f(f(a))) \;\&\; \sim Hf(f(f(a)))$	2 ∀E
11	$\sim Hf(f(f(a)))$	10 &E
12	$Df(a)f(a)$	8–11 ~ E
13	$(\forall x)Df(x)f(x)$	12 ∀I

Section 11.1E

5. Let $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$ be a quantificationally consistent set of sentences, none of which contains the constant **a**. Then there is some interpretation **I** on which every member of $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is true. Because $(\exists \mathbf{x})\mathbf{P}$ is true on **I**, we know that for any variable assignment **d**, there is a member **u** of the UD such that $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies **P** on **I**. Let **I′** be the interpretation that is just like **I** except that $\mathbf{I}'(\mathbf{a}) = \mathbf{u}$. Because **a** does not occur in $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$, it follows from 11.1.7 that every member of $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is true on **I′**.

On our assumption that $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies **P** on **I**, it follows from 11.1.6 that $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies **P** on **I′**. By the way that we have constructed **I′**, **u** is $\mathbf{I}'(\mathbf{a})$, and so $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ is $\mathbf{d}[\mathbf{I}'(\mathbf{a})/\mathbf{x}]$. From result 11.1.1, we therefore know that **d** satisfies $\mathbf{P}(\mathbf{a}/\mathbf{x})$ on **I′**. By 11.1.3, then, every variable assignment on **I′** satisfies $\mathbf{P}(\mathbf{a}/\mathbf{x})$, and so it is true on **I′**.

Every member of $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}, \mathbf{P}(\mathbf{a}/\mathbf{x})\}$ being true on **I′**, we conclude that the extended set is quantificationally consistent.

6. Assume that **I** is an interpretation on which each member of the UD is assigned to at least one individual constant and that every substitution instance of $(\forall \mathbf{x})\mathbf{P}$ is true on **I**. Now $(\forall \mathbf{x})\mathbf{P}$ is true on **I** if every variable assignment satisfies $(\forall \mathbf{x})\mathbf{P}$ and, by 11.1.3, if some variable assignment **d** satisfies $(\forall \mathbf{x})\mathbf{P}$. The latter is the case if for every member **u** of the UD, $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies **P**. Consider an arbitrary member **u** of the UD. By our assumption, $\mathbf{u} = \mathbf{I}(\mathbf{a})$ for some individual constant **a**. Also by assumption, $\mathbf{P}(\mathbf{a}/\mathbf{x})$ is true on **I**—so **d** satisfies $\mathbf{P}(\mathbf{a}/\mathbf{x})$. By 11.1.1, then, $\mathbf{d}[\mathbf{I}(\mathbf{a})/\mathbf{x}]$, which is $\mathbf{d}[\mathbf{u}/\mathbf{x}]$, satisfies **P**. We conclude that for every member **u** of the UD, $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies **P**, that **d** therefore satisfies $(\forall \mathbf{x})\mathbf{P}$, and that $(\forall \mathbf{x})\mathbf{P}$ is true on **I**.

Section 11.2E

4. To prove 11.2.5, we will make use of the following:

11.2.6. Let $\mathbf{t}_1$ and $\mathbf{t}_2$ be closed terms such that $\text{den}_{\mathbf{I,d}}(\mathbf{t}_1) = \text{den}_{\mathbf{I,d}}(\mathbf{t}_2)$, and let **t** be a term that contains $\mathbf{t}_1$. Then for any variable assignment **d**, and any term $\mathbf{t}(\mathbf{t}_2//\mathbf{t}_1)$ that results from replacing one or more occurrences of $\mathbf{t}_1$ in **t** with $\mathbf{t}_2$, $\text{den}_{\mathbf{I,d}}(\mathbf{t}(\mathbf{t}_2//\mathbf{t}_1)) = \text{den}_{\mathbf{I,d}}(\mathbf{t})$.

Proof. If $\mathbf{t}_1$ is **t**, then $\mathbf{t}(\mathbf{t}_2//\mathbf{t}_1)$ must be $\mathbf{t}_2$, and by assumption $\text{den}_{\mathbf{I,d}}(\mathbf{t}_1) = \text{den}_{\mathbf{I,d}}(\mathbf{t}_2)$.

For the case where **t** contains but is not identical to $\mathbf{t}_1$, we shall prove 11.2.6 by mathematical induction on the number of functors that occur in **t**—since **t** must be a complex term in this case.

Basis clause: If $\mathbf{t}$ contains one functor, then for any variable assignment $\mathbf{d}$, and any term $\mathbf{t}(\mathbf{t}_2//\mathbf{t}_1)$ that results from replacing one or more occurrences of $\mathbf{t}_1$ in $\mathbf{t}$ with $\mathbf{t}_2$, $\text{den}_{\mathbf{I},\mathbf{d}}(\mathbf{t}(\mathbf{t}_2//\mathbf{t}_1)) = \text{den}_{\mathbf{I},\mathbf{d}}(\mathbf{t})$.

Proof of basis clause: $\mathbf{t}$ has the form $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})$, where each $\mathbf{t}'_\mathbf{i}$ is a variable or constant. In this case, one or more of the $\mathbf{t}_\mathbf{i}$'s must be $\mathbf{t}_1$ and has been replaced by $\mathbf{t}_2$ to form $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})(\mathbf{t}_2//\mathbf{t}_1)$ and the remaining $\mathbf{t}_\mathbf{i}$'s are unchanged. In the former cases, by assumption we have $\text{den}_{\mathbf{I},\mathbf{d}}(\mathbf{t}_1) = \text{den}_{\mathbf{I},\mathbf{d}}(\mathbf{t}_2)$. So the denotations of the arguments at the corresponding positions in $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})$ and $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})(\mathbf{t}_2//\mathbf{t}_1)$ are identical, and therefore $\text{den}_{\mathbf{I},\mathbf{d}}(f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})) = \text{den}_{\mathbf{I},\mathbf{d}}(f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})(\mathbf{t}_2//\mathbf{t}_1))$.

Inductive step: If 11.2.6 holds for every term $\mathbf{t}$ that contains $\mathbf{k}$ or fewer functors, then it also holds for every term $\mathbf{t}$ that contains $\mathbf{k} + 1$ functors.

Proof of inductive step: Assume the inductive hypothesis for an arbitrary integer $\mathbf{k}$. We must show that 11.2.6 holds for every term $\mathbf{t}$ that contains $\mathbf{k} + 1$ functors. In this case, $\mathbf{t}$ has the form $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})$, where each $\mathbf{t}'_\mathbf{i}$ contains $\mathbf{k}$ or fewer functors and one or more of the $\mathbf{t}_\mathbf{i}$'s that is identical to or contains $\mathbf{t}_1$ has had one or more occurrences of $\mathbf{t}_1$ replaced by $\mathbf{t}_2$ to form $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})(\mathbf{t}_2//\mathbf{t}_1)$ and the remaining $\mathbf{t}_\mathbf{i}$'s are unchanged. In the former cases, it follows form the inductive hypothesis that the denotations of the arguments at the corresponding positions in $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})$ and $f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})(\mathbf{t}_2//\mathbf{t}_1)$ are identical, and therefore $\text{den}_{\mathbf{I},\mathbf{d}}(f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})) = \text{den}_{\mathbf{I},\mathbf{d}}(f(\mathbf{t}'_1, \ldots, \mathbf{t}'_\mathbf{n})(\mathbf{t}_2//\mathbf{t}_1))$.

We can now use 11.2.6 in the

Proof of 11.2.5: We shall prove only the first half of 11.2.5, since the second half is proved in the same way with minor modifications. Let $\mathbf{t}_1$ and $\mathbf{t}_2$ be closed terms and let $\mathbf{P}$ be a sentence that contains $\mathbf{t}_1$. If $\{\mathbf{t}_1 = \mathbf{t}_2, \mathbf{P}\}$ is quantificationally inconsistent then trivially $\{\mathbf{t}_1 = \mathbf{t}_2, \mathbf{P}\} \vDash \mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$.

If $\{\mathbf{t}_1 = \mathbf{t}_2, \mathbf{P}\}$ is quantificationally consistent, then let $\mathbf{I}$ be an interpretation on which both $\mathbf{t}_1 = \mathbf{t}_2$ and $\mathbf{P}$ are true and hence satisfied by every satisfaction assignment $\mathbf{d}$. We will show by mathematical induction on the number of occurrences of logical operators in a formula $\mathbf{P}$ that if $\mathbf{t}_1 = \mathbf{t}_2$ is satisfied by a satisfaction assignment $\mathbf{d}$ on an interpretation $\mathbf{I}$, then $\mathbf{P}$ is satisfied by $\mathbf{d}$ if and only if $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$ is satisfied by $\mathbf{d}$.

Basis clause: If $\mathbf{P}$ contains zero occurrences of logical operators and $\mathbf{t}_1 = \mathbf{t}_2$ is satisfied by a satisfaction assignment $\mathbf{d}$ on an interpretation $\mathbf{I}$ then $\mathbf{P}$ is satisfied by $\mathbf{d}$ if and only if $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$ is satisfied by $\mathbf{d}$ on $\mathbf{I}$.

Proof of basis clause: Since $\mathbf{P}$ contains $\mathbf{t}_1$, $\mathbf{P}$ must be either a formula of the form $\mathbf{A}\mathbf{t}'_1 \ldots \mathbf{t}'_\mathbf{n}$ or a formula of the form $\mathbf{t}_1 = \mathbf{t}_2$.

If $\mathbf{P}$ has the form $\mathbf{At'_1 \ldots t'_n}$ then $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$ is $\mathbf{At''_1 \ldots t''_n}$, where each $\mathbf{t''_i}$ is either $\mathbf{t'_i}$ or the result of replacing $\mathbf{t}_1$ in $\mathbf{t'_i}$ with $\mathbf{t}_2$. In the former case, $\text{den}_{\mathbf{I,d}}(\mathbf{t'_i}) = \text{den}_{\mathbf{I,d}}(\mathbf{t''_i})$ since $\mathbf{t''_i}$ is $\mathbf{t'_i}$. In the latter case, $\text{den}_{\mathbf{I,d}}(\mathbf{t'_i}) = \text{den}_{\mathbf{I,d}}(\mathbf{t''_i})$ by 11.2.6. So $<\text{den}_{\mathbf{I,d}}(\mathbf{t'_1}), \text{den}_{\mathbf{I,d}}(\mathbf{t'_2}), \ldots, \text{den}_{\mathbf{I,d}}(\mathbf{t'_n})> \; = \; <\text{den}_{\mathbf{I,d}}(\mathbf{t''_1}), \text{den}_{\mathbf{I,d}}(\mathbf{t''_2}), \ldots, \text{den}_{\mathbf{I,d}}(\mathbf{t''_n})>$ and so $<\text{den}_{\mathbf{I,d}}(\mathbf{t'_1}), \text{den}_{\mathbf{I,d}}(\mathbf{t'_2}), \ldots, \text{den}_{\mathbf{I,d}}(\mathbf{t'_n})>$ is a member of $\mathbf{I}(\mathbf{A})$ if and only if $<\text{den}_{\mathbf{I,d}}(\mathbf{t''_1}), \text{den}_{\mathbf{I,d}}(\mathbf{t''_2}), \ldots, \text{den}_{\mathbf{I,d}}(\mathbf{t''_n})>$ is a member of $\mathbf{I}(\mathbf{A})$. Consequently, $\mathbf{d}$ satisfies $\mathbf{At'_1 \ldots t'_n}$ if and only if $\mathbf{d}$ satisfies $\mathbf{At''_1 \ldots t''_n}$.

If $\mathbf{P}$ has the form $\mathbf{t'_1} = \mathbf{t'_2}$ then $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$ is $\mathbf{t''_1} = \mathbf{t''_2}$, where each $\mathbf{t''_i}$ is either $\mathbf{t'_i}$ or the result of replacing $\mathbf{t}_1$ in $\mathbf{t'_i}$ with $\mathbf{t}_2$. In the former case, $\text{den}_{\mathbf{I,d}}(\mathbf{t'_i}) = \text{den}_{\mathbf{I,d}}(\mathbf{t''_i})$ since $\mathbf{t''_i}$ is $\mathbf{t'_i}$. In the latter case, $\text{den}_{\mathbf{I,d}}(\mathbf{t'_i}) = \text{den}_{\mathbf{I,d}}(\mathbf{t''_i})$ by 11.2.6. It follows that $\text{den}_{\mathbf{I,d}}(\mathbf{t'_1}) = \text{den}_{\mathbf{I,d}}(\mathbf{t'_2})$ if and only if $\text{den}_{\mathbf{I,d}}(\mathbf{t''_1}) = \text{den}_{\mathbf{I,d}}(\mathbf{t''_2})$. Since $\mathbf{d}$ satisfies $\mathbf{t'_1} = \mathbf{t'_2}$ if and only if $\text{den}_{\mathbf{I,d}}(\mathbf{t'_1}) = \text{den}_{\mathbf{I,d}}(\mathbf{t'_2})$ and $\mathbf{d}$ satisfies $\mathbf{t''_1} = \mathbf{t''_2}$ if and only if $\text{den}_{\mathbf{I,d}}(\mathbf{t''_1}) = \text{den}_{\mathbf{I,d}}(\mathbf{t''_2})$, it follows that $\mathbf{d}$ satisfies $\mathbf{t'_1} = \mathbf{t'_2}$ if and only if it satisfies $\mathbf{t''_1} = \mathbf{t''_2}$.

Inductive step: If 11.2.5 is true of every formula $\mathbf{P}$ that contains $\mathbf{k}$ or fewer occurrences of logical operators then 11.2.5 is also true of every formula $\mathbf{P}$ that contains $\mathbf{k} + 1$ occurrences of logical operators.

Proof of inductive step: Assume that the inductive hypothesis holds for an arbitrary integer $\mathbf{k}$. Let $\mathbf{P}$ be a formula that contains $\mathbf{k} + 1$ logical operators. We must show that if $\mathbf{t}_1 = \mathbf{t}_2$ is satisfied by a satisfaction assignment $\mathbf{d}$ on an interpretation $\mathbf{I}$ then $\mathbf{P}$ is satisfied by $\mathbf{d}$ if and only if $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$ is also satisfied by $\mathbf{d}$. We shall show this by considering each form that $\mathbf{P}$ might have.

Case 1. $\mathbf{P}$ is a formula of the form $\sim \mathbf{Q}$. Then $\mathbf{P}$ is satisfied by $\mathbf{d}$ if and only if $\mathbf{Q}$ is not satisfied by $\mathbf{d}$. Since $\mathbf{Q}$ contains $\mathbf{k}$ logical operators, it follows by the inductive hypothesis that $\mathbf{Q}$ is not satisfied by $\mathbf{d}$ if and only if $\mathbf{Q}(\mathbf{t}_2//\mathbf{t}_1)$ is not satisfied by $\mathbf{d}$, and this is the case if and only if $\sim \mathbf{Q}(\mathbf{t}_2//\mathbf{t}_1)$, which is $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$, is satisfied by $\mathbf{d}$.

Cases 2–5. $\mathbf{P}$ has one of the forms $(\mathbf{Q} \;\&\; \mathbf{R})$, $(\mathbf{Q} \vee \mathbf{R})$, $(\mathbf{Q} \supset \mathbf{R})$, or $(\mathbf{Q} \equiv \mathbf{R})$. Similar to case 1.

Case 6. $\mathbf{P}$ has the form $(\forall \mathbf{x})\mathbf{Q}$. Then $\mathbf{P}$ is satisfied by $\mathbf{d}$ if and only if every variable assignment $\mathbf{d}'$ that is like $\mathbf{d}$ except possibly in the value assigned to $\mathbf{x}$ satisfies $\mathbf{Q}$. Since $\mathbf{t}_1$ and $\mathbf{t}_2$ are closed terms, every such variable assignment $\mathbf{d}'$ will satisfy $\mathbf{t}_1 = \mathbf{t}_2$ since $\text{den}_{\mathbf{I,d}}(\mathbf{t}_1) = \text{den}_{\mathbf{I,d'}}(\mathbf{t}_1)$ and $\text{den}_{\mathbf{I,d}}(\mathbf{t}_2) = \text{den}_{\mathbf{I,d'}}(\mathbf{t}_2)$ by 11.2.2. Because $\mathbf{Q}$ contains $\mathbf{k}$ occurrences of logical operators, it follows by the inductive hypothesis that every such variable assignment $\mathbf{d}'$ will satisfy $\mathbf{Q}$ if and only if it also satisfies $\mathbf{Q}(\mathbf{t}_2//\mathbf{t}_1)$, and every such variable assignment $\mathbf{d}'$ will satisfy $\mathbf{Q}(\mathbf{t}_2//\mathbf{t}_1)$ if and only if $\mathbf{d}$ satisfies $(\forall \mathbf{x})\mathbf{Q}(\mathbf{t}_2//\mathbf{t}_1)$, which is $\mathbf{P}(\mathbf{t}_2//\mathbf{t}_1)$ ($\mathbf{t}_1$, being a closed term, is not the variable $\mathbf{x}$).

Case 7. $\mathbf{P}$ has the form $(\exists \mathbf{x})\mathbf{Q}$. Similar to case 6.

Section 11.3E

1.a. Assume that an argument of *PL* is valid in *PD*. Then the conclusion is derivable in *PD* from the set consisting of the premises. By Metatheorem 11.3.1, it follows that the conclusion is quantificationally entailed by the set consisting of the premises. Therefore the argument is quantificationally valid.

b. Assume that a sentence **P** is a theorem in *PD*. Then $\varnothing \vdash \mathbf{P}$. So $\varnothing \vDash \mathbf{P}$, by Metatheorem 11.3.1, and **P** is quantificationally true.

2. Our induction will be on the number of occurrences of *logical operators* in **P**, for we must now take into account the quantifiers as well as the truth-functional connectives.

Basis clause: Thesis 11.3.4 holds for every atomic formula of *PL*.

> **Proof:** Assume that **P** is an atomic formula and that **Q** is a subformula of **P**. Then **P** and **Q** are identical. For any formula $\mathbf{Q}_1$, then, $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is simply $\mathbf{Q}_1$. It is trivial that the thesis holds in this case.

Inductive step: Let **P** be a formula with **k** + 1 occurrences of logical operators, let **Q** be a subformula of **P**, and let $\mathbf{Q}_1$ be a formula related to **Q** as stipulated. Assume (the inductive hypothesis) that 11.3.4 holds for every formula with **k** or fewer occurrences of logical operators. We now establish that 11.3.4 holds for **P** as well. Suppose first that **Q** and **P** are identical. In this case, that 11.3.4 holds for **P** and $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is established as in the proof of the basis clause. So assume that **Q** is a subformula of **P** that is not identical with **P** (in which case we say that **Q** is a *proper subformula* of **P**). We consider each form that **P** may have.

(i) P is of the form ~ **R**. Since **Q** is a proper subformula of **P**, **Q** is a subformula of **R**. Therefore $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is $\sim [\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$. Since **R** has fewer than **k** + 1 occurrences of logical operators, it follows from the inductive hypothesis that, on any interpretation, a variable assignment satisfies **R** if and only if it satisfies $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$. Since an assignment satisfies a formula if and only if it fails to satisfy the negation of the formula, it follows that on any interpretation a variable assignment satisfies ~ **R** if and only if it satisfies $\sim [\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$.

(ii)–(v) P is of the form **R** & **S**, $\mathbf{R} \vee \mathbf{S}$, $\mathbf{R} \supset \mathbf{S}$, or $\mathbf{R} \equiv \mathbf{S}$. These cases are handled similarly to case (ii) in the inductive proof of Lemma 6.1 (in Chapter 6), with obvious adjustments as in case (i).

(vi) P is of the form $(\forall \mathbf{x})\mathbf{R}$. Since **Q** is a proper subformula of **P**, **Q** is a subformula of **R**. Therefore $[\mathbf{P}](\mathbf{Q}_1//\mathbf{Q})$ is $(\forall \mathbf{x})[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$. Since **R** has fewer than **k** + 1 occurrences of logical operators, it follows, by the inductive hypothesis, that on any interpretation a variable assignment satisfies **R** if and only if that assignment satisfies $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$. Now $(\forall \mathbf{x})\mathbf{R}$ is satisfied by a variable assignment **d** if and only if for each member **u** of the UD, $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies **R**. The latter is the case just in case $[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$ is satisfied by every variant $\mathbf{d}[\mathbf{u}/\mathbf{x}]$. And this is the case if and only if $(\forall \mathbf{x})[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$ is satisfied by **d**. Therefore on any interpretation $(\forall \mathbf{x})[\mathbf{R}$ is satisfied by a variable assignment if and only if $(\forall \mathbf{x})[\mathbf{R}](\mathbf{Q}_1//\mathbf{Q})$ is satisfied by that assignment.

(vii) P is of the form $(\exists \mathbf{x})\mathbf{R}$. This case is similar to case (vi).

3. $\mathbf{Q}_{\mathbf{k}+1}$ is justified at position $\mathbf{k} + 1$ by Quantifier Negation. Then $\mathbf{Q}_{\mathbf{k}+1}$ is derived as follows:

$\mathbf{h}$	$\mathbf{S}$	
$\mathbf{k} + 1$	$\mathbf{Q}_{\mathbf{k}+1}$	$\mathbf{h}$ QN

where some component $\mathbf{R}$ of $\mathbf{S}$ has been replaced by a component $\mathbf{R}_1$ to obtain $\mathbf{Q}_{\mathbf{k}+1}$ and the four forms that $\mathbf{R}$ and $\mathbf{R}_1$ may have are

$\mathbf{R}$ is	$\mathbf{R}_1$ is
$\sim (\forall \mathbf{x})\mathbf{P}$	$(\exists \mathbf{x}) \sim \mathbf{P}$
$(\exists \mathbf{x}) \sim \mathbf{P}$	$\sim (\forall \mathbf{x})\mathbf{P}$
$\sim (\exists \mathbf{x})\mathbf{P}$	$(\forall \mathbf{x}) \sim \mathbf{P}$
$(\forall \mathbf{x}) \sim \mathbf{P}$	$\sim (\exists \mathbf{x})\mathbf{P}$

Whichever pair $\mathbf{R}$ and $\mathbf{R}_1$ constitute, the two sentences contain exactly the same nonlogical constants. We first establish that on any interpretation variable assignment $\mathbf{d}$ satisfies $\mathbf{R}$ if and only if $\mathbf{d}$ satisfies $\mathbf{R}_1$.

(i) Either $\mathbf{R}$ is $\sim (\forall \mathbf{x})\mathbf{P}$ and $\mathbf{R}_1$ is $(\exists \mathbf{x}) \sim \mathbf{P}$, or $\mathbf{R}$ is $(\exists \mathbf{x}) \sim \mathbf{P}$ and $\mathbf{R}_1$ is $\sim (\forall \mathbf{x})\ \mathbf{P}$. Assume that a variable assignment $\mathbf{d}$ satisfies $\sim (\forall \mathbf{x})\mathbf{P}$. Then $\mathbf{d}$ does not satisfy $(\forall \mathbf{x})\mathbf{P}$. There is then at least one variant $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ that does not satisfy $\mathbf{P}$. Hence $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies $\sim \mathbf{P}$. It follows that $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies $(\exists \mathbf{x}) \sim \mathbf{P}$. Now assume that a variable assignment $\mathbf{d}$ satisfies $(\exists \mathbf{x}) \sim \mathbf{P}$. Then some variant $\mathbf{d}[\mathbf{u}/\mathbf{x}]$ satisfies $\sim \mathbf{P}$. This variant does not satisfy $\mathbf{P}$. Therefore $\mathbf{d}$ does not satisfy $(\forall \mathbf{x})\mathbf{P}$ and does satisfy $\sim (\forall \mathbf{x})\mathbf{P}$.

(ii) Either $\mathbf{R}$ is $\sim (\exists \mathbf{x})\mathbf{P}$ and $\mathbf{R}_1$ is $(\forall \mathbf{x}) \sim \mathbf{P}$, or $\mathbf{R}$ is $(\forall \mathbf{x}) \sim \mathbf{P}$ and $\mathbf{R}_1$ is $\sim (\exists \mathbf{x})\mathbf{P}$. This case is similar to case (i).

$\mathbf{R}$ and $\mathbf{R}_1$ contain the same nonlogical symbols and variables, so it follows, by 11.3.4 (Exercise 2), that $\mathbf{S}$ is satisfied by a variable assignment if and only if $\mathbf{Q}_{\mathbf{k}+1}$ is satisfied by that assignment. So on any interpretation $\mathbf{S}$ and $\mathbf{Q}_{\mathbf{k}+1}$ have the same truth-value.

By the inductive hypothesis, $\Gamma_{\mathbf{k}} \vDash \mathbf{S}$. But $\Gamma_{\mathbf{k}}$ is a subset of $\Gamma_{\mathbf{k}+1}$, and so $\Gamma_{\mathbf{k}+1} \vDash \mathbf{S}$, by 11.3.2. Since $\mathbf{S}$ and $\mathbf{Q}_{\mathbf{k}+1}$ have the same truth-value on any interpretation, it follows that $\Gamma_{\mathbf{k}+1} \vDash \mathbf{Q}_{\mathbf{k}+1}$.

Section 11.4E

2. Assume that $\Gamma \cup \{\sim \mathbf{P}\}$ is inconsistent in *PD*. Then there is a derivation of the following sort, where $\mathbf{Q}_1, \ldots, \mathbf{Q}_{\mathbf{n}}$ are members of Γ:

1	$\mathbf{Q}_1$	Assumption
.	.	
$\mathbf{n}$	$\mathbf{Q}_{\mathbf{n}}$	Assumption
$\mathbf{n} + 1$	$\sim \mathbf{P}$	Assumption
.	.	
$\mathbf{m}$	$\mathbf{S}$	
.	.	
$\mathbf{p}$	$\sim \mathbf{S}$	

We construct a new derivation as follows:

1	$\mathbf{Q}_1$	Assumption
.	.	
n	$\mathbf{Q}_\mathbf{n}$	Assumption
n + 1	~ **P**	Assumption
.	.	A / ~ **E**
m	**S**	
.	.	
p	~ **S**	
p + 1	**P**	**n** + 1 − **p** ~ **E**

where lines 1 to **p** are as in the original derivation, except that ~ **P** is now an auxiliary assumption. This shows that $\Gamma \vDash \mathbf{P}$.

3.a. Assume that an argument of *PL* is quantificationally valid. Then the set consisting of the premises quantificationally entails the conclusion. By Metatheorem 11.4.1, the conclusion is derivable from that set in *PD*. Therefore the argument is valid in *PD*.

b. Assume that a sentence **P** is quantificationally true. Then $\varnothing \vDash \mathbf{P}$. By Metatheorem 11.4.1, $\varnothing \vDash \mathbf{P}$. So **P** is a theorem in *PD*.

4. We shall associate with each symbol of *PL* a numeral as follows. With each symbol of *PL* that is a symbol of *SL*, associate the two-digit numeral that is associated with that symbol in the enumeration of Section 6.4. With the symbol ′ (the prime) associate the numeral ‘66’. With the nonsubscripted lowercase letters ‘a’, ‘b’, . . . , ‘z’, associate the numerals ‘67’, ‘68’, . . . , ‘92’, respectively. With the symbols ‘∀’ and ‘∃’ associate the numerals ‘93’ and ‘94’, respectively. (Note that the numerals ‘66’ to ‘94’ are not associated with any symbol of *SL*.) We then associate with each sentence of *PL* the numeral that consists of the associated numerals of each of the symbols that occur in the sentence, in the order in which the symbols occur. We now enumerate the sentences of *PL* by letting the first sentence be the sentence whose numeral designates a number that is smaller than the number designated by any other sentence’s associated numeral; the second sentence is the sentence whose numeral designates the next largest number designated by the associated numeral of any sentence; and so on.

5. Assume that $\Gamma \vdash \mathbf{P}$. Then there is a derivation

1	$\mathbf{Q}_1$
.	.
n	$\mathbf{Q}_\mathbf{n}$
.	.
m	**P**

where $\mathbf{Q}_1$, . . . , $\mathbf{Q}_\mathbf{n}$ are all members of Γ. The primary assumptions are all members of any superset $\Gamma' \vdash$ of Γ, and so $\Gamma' \vdash \mathbf{P}$ as well.

6.a. Assume that **a** does not occur in any member of the set $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$ and that the set is consistent in *PD*. Assume, contrary to what we want to prove, that $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}, \mathbf{P}(\mathbf{a}/\mathbf{x})\}$ is *in*consistent in *PD*. Then there is a derivation of the sort

$$\begin{array}{r|l}
1 & \mathbf{Q}_1 \\
. & . \\
\mathbf{n} & \mathbf{Q_n} \\
\mathbf{n}+1 & (\exists \mathbf{x})\mathbf{P} \\
\mathbf{n}+2 & \mathbf{P}(\mathbf{a}/\mathbf{x}) \\ \cline{2-2}
\mathbf{m} & \mathbf{R} \\
. & . \\
\mathbf{p} & \sim \mathbf{R}
\end{array}$$

where $\mathbf{Q}_1, \ldots, \mathbf{Q_n}$ are all members of Γ. We may convert this into a derivation showing that $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is inconsistent in *PD*, contradicting our initial assumption:

$$\begin{array}{r|l|l|ll}
1 & \multicolumn{3}{l}{\mathbf{Q}_1} & \\
. & \multicolumn{3}{l}{.} & \\
\mathbf{n} & \multicolumn{3}{l}{\mathbf{Q_n}} & \\
\mathbf{n}+1 & \multicolumn{3}{l}{(\exists \mathbf{x})\mathbf{P}} & \\ \cline{2-4}
\mathbf{n}+2 & & \multicolumn{2}{l}{\mathbf{P}(\mathbf{a}/\mathbf{x})} & \\ \cline{3-4}
\mathbf{n}+3 & & & (\exists \mathbf{x})\mathbf{P} & \\ \cline{4-4}
. & & & . & \\
\mathbf{m}+1 & & & \mathbf{R} & \\
. & & & . & \\
\mathbf{p}+1 & & & \sim \mathbf{R} & \\
\mathbf{p}+2 & & \multicolumn{2}{l}{\sim (\exists \mathbf{x})\mathbf{P}} & \mathbf{n}+3-\mathbf{p}+1 \quad \sim \mathrm{I} \\
\mathbf{p}+3 & \multicolumn{3}{l}{\sim (\exists \mathbf{x})\mathbf{P}} & \mathbf{n}+2-\mathbf{p}+2 \quad \exists\mathrm{E} \\
\mathbf{p}+4 & \multicolumn{3}{l}{(\exists \mathbf{x})\mathbf{P}} & \mathbf{n}+1 \quad \mathrm{R}
\end{array}$$

(Note that use of ∃E is legitimate at line $\mathbf{p} + 3$ because **a**, by our initial hypothesis, does not occur in $(\exists \mathbf{x})\mathbf{P}$ or in any member of Γ.)

We conclude that if the set $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is consistent in *PD* and **a** does not occur in any member of that set, then $\Gamma \cup \{(\exists \mathbf{x})\mathbf{P}(\mathbf{a}/\mathbf{x})\}$ is also consistent in *PD*.

b. Let Γ* be constructed as in our proof of Lemma 11.4.4. Assume that $(\exists \mathbf{x})\mathbf{P}$ is a member of Γ* and that $(\exists \mathbf{x})\mathbf{P}$ is the **i**th sentence in our enumeration of the sentences of *PL*. Then, by the way each member of the infinite sequence $\Gamma_1, \Gamma_2, \Gamma_3, \ldots$ is constructed, $\Gamma_{\mathbf{i}+1}$ contains $(\exists \mathbf{x})\mathbf{P}$ and a substitution instance of $(\exists \mathbf{x})\mathbf{P}$ if $\Gamma_{\mathbf{i}} \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is consistent in *PD*. Since each member of the infinite sequence is consistent in *PD*, $\Gamma_{\mathbf{i}}$ is consistent to *PD*. So assume that $\Gamma_{\mathbf{i}} \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is inconsistent in *PD*. Then, since we assumed that $\mathbf{P_i}$, that is, $(\exists \mathbf{x})\mathbf{P}$, is a member of Γ* and since every member of $\Gamma_{\mathbf{i}}$ is a member of Γ*, it follows that Γ* is inconsistent in *PD*. But this contradicts our original assumption, and so $\Gamma_{\mathbf{i}} \cup \{(\exists \mathbf{x})\mathbf{P}\}$ is consistent in *PD*. Hence $\Gamma_{\mathbf{i+1}}$ is $\Gamma_{\mathbf{i}} \cup \{(\exists \mathbf{x})\mathbf{P}, \mathbf{P}(\mathbf{a}/\mathbf{x})\}$ for some constant **a**, and so some substitution instance of $(\exists \mathbf{x})\mathbf{P}$ is a member of $\Gamma_{\mathbf{i+1}}$ and thus of Γ*.

7. We shall prove that the sentence at each position **i** in the new derivation can be justified by the same rule that was used at position **i** in the original derivation.

Basis clause: Let **i** = 1. The sentence at position 1 of the original derivation is an assumption, and so the sentence at position 1 of the new sequence can be justified similarly.

Inductive step: Assume (the inductive hypothesis) that at every position **i** prior to position **k** + 1, the new sequence contains a sentence that may be justified by the rule justifying the sentence at position **i** of the original derivation. We now prove that the sentence at position **k** + 1 of the new sequence can be justified by the rule justifying the sentence at position **k** + 1 of the original derivation. We shall consider the rules by which the sentence at position **k** + 1 of the original derivation could have been justified:

1. **P** is justified at position **k** + 1 by Assumption. Obviously, **P*** can be justified by Assumption at position **k** + 1 of the new sequence.

2. **P** is justified at position **k** + 1 by Reiteration. Then **P** occurs at an accessible earlier position in the original derivation. Therefore **P*** occurs at an accessible earlier position in the new sequence, so **P*** can be justified at position **k** + 1 by Reiteration.

3. **P** is a conjunction **Q** & **R** justified at position **k** + 1 by Conjunction Introduction. Then the conjuncts **Q** and **R** of **P** occur at accessible earlier positions in the original derivation. Therefore **Q*** and **R*** occur at accessible earlier positions in the new sequence. So **P***, which is just **Q*** & **R***, can be justified at position **k** + 1 by Conjunction Introduction.

4–12. **P** is justified by one of the other truth-functional connective introduction or elimination rules. These cases are as straightforward as case 3, so we move on to the quantifier rules.

13. **P** is a sentence $\mathbf{Q}(\mathbf{a}/\mathbf{x})$ justified at position **k** + 1 by ∀E, appealing to an accessible earlier position with $(\forall \mathbf{x})\mathbf{Q}$. Then $(\forall \mathbf{x})\mathbf{Q}^*$ occurs at the accessible earlier position of the new sequence, and $\mathbf{Q}(\mathbf{a}/\mathbf{x})^*$ occurs at position **k** + 1. But $\mathbf{Q}(\mathbf{a}/\mathbf{x})^*$ is just a substitution instance of $(\forall \mathbf{x})\mathbf{Q}^*$. So $\mathbf{Q}(\mathbf{a}/\mathbf{x})^*$ can be justified at position **k** + 1 by ∀E.

14. **P** is a sentence $(\exists \mathbf{x})\mathbf{Q}$ and is justified at position **k** + 1 by ∃I. This case is similar to case 13.

15. **P** is a sentence $(\forall \mathbf{x})\mathbf{Q}$ and is justified at position **k** + 1 by ∀I. Then some substitution instance occurs at an accessible earlier position **j**, where **a** is

a constant that does not occur in any open assumption prior to position $\mathbf{k} + 1$ or in $(\forall \mathbf{x})\mathbf{Q}$. $\mathbf{Q}(\mathbf{a}/\mathbf{x})^*$ and $(\forall \mathbf{x})\mathbf{Q}^*$ occur at positions $\mathbf{j}$ and $\mathbf{k} + 1$ of the new sequence. $\mathbf{Q}(\mathbf{a}/\mathbf{x})^*$ is a substitution instance of $(\forall \mathbf{x})\mathbf{Q}^*$. The instantiating constant $\mathbf{a}$ in $\mathbf{Q}(\mathbf{a}/\mathbf{x})$ is some $\mathbf{a_i}$, and so the instantiating constant in $\mathbf{Q}(\mathbf{a}/\mathbf{x})^*$ is $\mathbf{b_i}$. Since $\mathbf{a_i}$ did not occur in any open assumption before position $\mathbf{k} + 1$ or in $(\forall \mathbf{x})\mathbf{Q}$ in the original derivation and $\mathbf{b_i}$ does not occur in the original derivation, $\mathbf{b_i}$ does not occur in any open assumption prior to position $\mathbf{k} + 1$ of the new sequence or in $(\forall \mathbf{x})\mathbf{Q}^*$. So $(\forall \mathbf{x})\mathbf{Q}^*$ can be justified by ∀I at position $\mathbf{k} + 1$ in the new sequence.

16. $\mathbf{P}$ is justified at position $\mathbf{k} + 1$ by ∃E. This case is similar to case 15.

Since every sentence in the new sequence can be justified by a rule of *PD*, it follows that the new sequence is indeed a derivation of *PD*.

10. We required that Γ^* be ∃-complete so that we could construct an interpretation I^* for which we could *prove* that every member of Γ^* is true on I^*. In requiring that Γ be ∃-complete in addition to being maximally consistent in *PD*, we were guaranteed that Γ^* had property g of sets that are both maximally consistent in *PD* and ∃-complete; and we used this fact in case 7 of the proof that every member of Γ^* is true on I^*.

11. To prove that PD^* is complete for predicate logic, it will suffice to show that with $\forall\text{E}^*$ instead of ∀E, every set Γ^* of PD^* that is both maximally consistent in PD^* and ∃-complete has property f (i.e., $(\forall \mathbf{x})\mathbf{P} \in \Gamma^*$ if and only if for every constant $\mathbf{a}$, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma^*$). For the properties a to e and g can be shown to characterize such sets by appealing to the rules of PD^* that are rules of *PD*. Here is our proof:

> **Proof**: Assume that $(\forall \mathbf{x})\mathbf{P} \in \Gamma^*$. Then, since $\{(\forall \mathbf{x})\mathbf{P}\} \vdash \sim (\exists \mathbf{x}) \sim \mathbf{P}$ by $\forall\text{E}^*$, it follows from 11.3.3 that $\sim (\exists \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$. Then $(\exists \mathbf{x}) \sim \mathbf{P} \notin \Gamma^*$, by a. Assume that for some substitution instance $\mathbf{P}(\mathbf{a}/\mathbf{x})$ of $(\forall \mathbf{x})\mathbf{P}$, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \notin \Gamma^*$. Then, by a, $\sim \mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma^*$. Since $\{\sim \mathbf{P}(\mathbf{a}/\mathbf{x})\} \vdash (\exists \mathbf{x}) \sim \mathbf{P}$ (without use of ∀E), it follows that $(\exists \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$. But we have just shown that $(\exists \mathbf{x}) \sim \mathbf{P} \notin \Gamma^*$. Hence, if $(\forall \mathbf{x})\mathbf{P} \in \Gamma^*$, then every substitution instance $\mathbf{P}(\mathbf{a}/\mathbf{x})$ of $(\forall \mathbf{x})\mathbf{P}$ is a member of Γ^*.
>
> Now assume that $(\forall \mathbf{x})\mathbf{P} \notin \Gamma^*$. Then, by a, $\sim (\forall \mathbf{x})\mathbf{P} \in \Gamma^*$. But then, since $\{\sim (\forall \mathbf{x})\mathbf{P}\} \vdash (\exists \mathbf{x}) \sim \mathbf{P}$ (without use of ∀E), it follows that $(\exists \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$. Since Γ^* is ∃-complete, some substitution instance $\sim \mathbf{P}(\mathbf{a}/\mathbf{x})$ of $(\exists \mathbf{x}) \sim \mathbf{P}$ is a member of Γ^*. By a, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \notin \Gamma^*$.

13. Assume that some sentence $\mathbf{P}$ is not quantificationally false. Then $\mathbf{P}$ is true on at least one interpretation, so $\{\mathbf{P}\}$ is quantificationally consistent. Now suppose that $\{\mathbf{P}\}$ is inconsistent in *PD*. Then some sentences $\mathbf{Q}$ and $\sim \mathbf{Q}$ are derivable from $\{\mathbf{P}\}$ in *PD*. By Metatheorem 11.3.1, it follows that $\{\mathbf{P}\} \vDash \mathbf{Q}$ and $\{\mathbf{P}\} \vDash \sim \mathbf{Q}$. But then $\mathbf{P}$ cannot be true on any interpretation, contrary to our assumption. So {**P**} is consistent in *PD*. By 11.4.3 and 11.4.4 {$\mathbf{P_e}$}—the set resulting from doubling the subscript of every individual constant in **P**—is a subset of a set Γ* that is both maximally consistent in *PD* and ∃-complete. It follows from Lemma 11.4.8 that Γ* is quantificationally consistent. But, in proving 11.4.8, we actually showed more—for the characteristic interpretation **I*** that we constructed for Γ* has the set of positive integers as UD. Hence every member of Γ* is true on some interpretation with the set of positive integers as UD, and thus $\mathbf{P_e}$ is true on some interpretation with the set of positive integers as UD. **P** can also be shown true on some interpretation with that UD, using 11.1.13.

16. We shall prove 11.4.1 by mathematical induction on the number of functors occurring in **t**.

> *Basis clause:* 11.4.1 holds of every complex closed term that contains 1 occurrence of a functor.
>
> **Proof of basis clause:** If **t** contains 1 functor then **t** is $f(\mathbf{t}_1, \ldots, \mathbf{t_n})$, where each $\mathbf{t_i}$ is a constant. Let **a** be the alphabetically earliest constant such that $f(\mathbf{t}_1, \ldots, \mathbf{t_n}) = \mathbf{a}$ is a member of Γ*. It follows from clause 4 of the definition of **I*** that $\mathbf{I}^*(f)$ includes $<\mathbf{I}^*(\mathbf{t}_1), \ldots, \mathbf{I}^*(\mathbf{t_n}), \mathbf{I}^*(\mathbf{a})>$ and so $\text{den}_{\mathbf{I}^*,\mathbf{d}}\ (f(\mathbf{t}_1, \ldots, \mathbf{t_n})) = \mathbf{I}^*(\mathbf{a})$.
>
> *Inductive step:* If 11.4.1 holds of every complex closed term that contains **k** or fewer occurrences of functors, then 11.4.1 holds of every complex closed term that contains **k** occurrences of functors.
>
> **Proof of inductive step:** Assume the inductive hypothesis: that 11.4.1 holds of every complex closed term that contains **k** or fewer occurrences of functors. Let **t** be a term that contains **k** + 1 occurrences of functors; we will show that 11.4.1 holds of **t** as well.
>
> **t** has the form $f(\mathbf{t}_1, \ldots, \mathbf{t_n})$, where each $\mathbf{t_i}$ is a closed term containing **k** or fewer occurrences of functors. Let **a** be the alphabetically earliest constant such that $f(\mathbf{t}_1, \ldots, \mathbf{t_n}) = \mathbf{a}$ is a member of Γ*. It follows from the inductive hypothesis that for each $\mathbf{t_i}$, $\text{den}_{\mathbf{I}^*,\mathbf{d}}(\mathbf{t_i}) = \mathbf{I}^*(\mathbf{a_i})$, where $\mathbf{a_i}$ is the alphabetically earliest constant such that $\mathbf{t_i} = \mathbf{a_i}$ is a member of Γ*. It follows from property (**i**) of maximally consistent, ∃-complete sets that $f(\mathbf{a}_1, \ldots, \mathbf{a_n}) = \mathbf{a}$ is a member of Γ*, and it follows from clause 4 of the definition of **I*** that $\mathbf{I}^*(f)$ includes $<\mathbf{I}^*(\mathbf{a}_1), \ldots, \mathbf{I}^*(\mathbf{a_n}), \mathbf{I}^*(\mathbf{a})>$ So $\text{den}_{\mathbf{I}^*,\mathbf{d}}(f(\mathbf{t}_1, \ldots, \mathbf{t_n})) = \text{den}_{\mathbf{I}^*,\mathbf{d}}(f(\mathbf{a}_1, \ldots, \mathbf{a_n})) = \mathbf{I}^*(\mathbf{a})$.

17. Consider the sentence '(∀x)(∀y)x = y'. This sentence is not quantificationally false; it is true on every interpretation with a one-member UD. In addition, however, it is true on *only* those interpretations that have one-member UDs. (This is because for any variable assignment and any members $\mathbf{u}_1$ and $\mathbf{u}_2$ of a UD, $\mathbf{d}[\mathbf{u}_1/x, \mathbf{u}_2/y]$ satisfies 'x = y' as required for the truth of '(∀x) (∀y)x = y' if and only if $\mathbf{u}_1$ and $\mathbf{u}_2$ are the same object.) So there can be no interpretation with the set of positive integers as UD on which the sentence is true.

Section 11.5E

2.a. Assume that for some sentence **P**, {**P**} has a closed truth-tree. Then, by 11.5.1, {**P**} is quantificationally inconsistent. Hence there is no interpretation on which **P**, the sole member of {**P**}, is true. Therefore **P** is quantificationally false.

b. Assume that for some sentence **P**, {~ **P**} has a closed truth-tree. Then, by 11.5.1, {~ **P**} is quantificationally inconsistent. Hence there is no interpretation on which ~ **P** is true. So **P** is true on every interpretation; that is, **P** is quantificationally true.

d. Assume that $\Gamma \cup$ {~ **P**} has a closed truth-tree. Then, by 11.5.1, $\Gamma \cup$ {~ **P**} is quantificationally inconsistent. Hence there is no interpretation on which every member of Γ is true and ~ **P** is also true. That is, there is no interpretation on which every member of Γ is true and **P** is false. But then $\Gamma \vDash$ **P**.

3.a. **P** is obtained from ~ ~ **P** by ~ ~ D. It is straightforward that {~ ~ **P**} $\vDash$ **P**.

d. **P** or ~ **Q** is obtained from ~ (**P** $\supset$ **Q**) by ~ $\supset$D. On any interpretation on which ~ (**P** $\supset$ **Q**) is true, **P** $\supset$ **Q** is false—hence **P** is true and **Q** is false. But, if **Q** is false, then ~ **Q** is true. Thus {~ (**P** $\supset$ **Q**)} $\vDash$ **P**, and {~ (**P** $\supset$ **Q**)} $\vDash$ ~ **Q**.

e. **P**(**a**/**x**) is obtained from ($\forall$**x**)**P** by $\forall$D. It follows, from 11.1.4, that {($\forall$**x**)**P**} $\vDash$ **P**(**a**/**x**).

4.a. ~ **P** and ~ **Q** are obtained from ~ (**P** & **Q**) by ~ &D. On any interpretation on which ~ (**P** & **Q**) is true, **P** & **Q** is false. But then either **P** is false, or **Q** is false. Hence on such an interpretation either ~ **P** is true, or ~ **Q** is true.

5. The path is extended to form two paths to level **k** + 1 as a result of applying one of the branching rules $\equiv$D or ~ $\equiv$D to a sentence **P** on $\Gamma_{\mathbf{k}}$. We consider four cases.

a. Sentences **P** and ~ **P** are entered at level **k** + 1 as the result of applying $\equiv$D to a sentence **P** $\equiv$ **Q** on $\Gamma_{\mathbf{k}}$. On any interpretation on which **P** $\equiv$ **Q** is true, so is either **P** or ~ **P**. Therefore either **P** and all the sentences on $\Gamma_{\mathbf{k}}$ are true on $\mathbf{I}_{\Gamma_{\mathbf{k}}}$, which is a path variant of **I** for the new path containing **P**, or ~ **P** and all the sentences on $\Gamma_{\mathbf{k}}$ are true on $\mathbf{I}_{\Gamma_{\mathbf{k}}}$, which is a path variant of **I** for the new path containing ~ **P**.

b. Sentence **Q** (or ~ **Q**) is entered at level **k** + 1 as the result of applying $\equiv$D to a sentence **P** $\equiv$ **Q** on $\Gamma_{\mathbf{k}}$. Then **P** (or ~ **P**) occurs on $\Gamma_{\mathbf{k}}$ at level **k** (application of $\equiv$D involves making entries at two levels, and **Q** and ~ **Q** are entries made on the second of these levels). Since {**P** $\equiv$ **Q**, **P**} quantificationally entails **Q** (and {**P** $\equiv$ **Q**, ~ **P**} quantificationally entails ~ **Q**), it follows that **Q** and all the sentences on $\Gamma_{\mathbf{k}}$ (~ **Q** and all the sentences on $\Gamma_{\mathbf{k}}$) are all true on $\mathbf{I}_{\Gamma_{\mathbf{k}}}$, which is a path variant of **I** for the new path containing **Q** (~ **Q**).

c. Sentences **P** and ~ **P** are entered at level **k** + 1 as the result of applying ~ $\equiv$D to a sentence ~ (**P** $\equiv$ **Q**) on $\Gamma_{\mathbf{k}}$. This case is similar to (a).

d. Sentence **Q** (or ~ **Q**) is entered at level **k** + 1 as the result of applying ~ $\equiv$D to a sentence ~ (**P** $\equiv$ **Q**) on $\Gamma_{\mathbf{k}}$. This case is similar to (b).

6. Yes. Dropping a rule would not make the method unsound, for, with the remaining rules, it would still follow that if a branch on a tree for a set Γ closes, then Γ is quantificationally inconsistent. That is, the remaining rules would still be consistency-preserving.

7. In proving that the tree method for *SL* is sound, there are obvious adjustments that must be made in the proof of Metatheorem 11.5.1. First, not all the tree rules for *PL* are tree rules for *SL*. In proving Lemma 11.5.2, then, we take only the tree rules for *SL* into consideration. And in the case of *SL* we would be proving that certain sets are truth-functionally consistent or inconsistent, rather than quantificationally consistent or inconsistent. The basic semantic concept for *SL* is that of a truth-value assignment, rather than an interpretation. With these stipulations, the proof of Metatheorem 11.5.1 can be converted straight-forwardly into a proof of the parallel metatheorem for *SL*.

Section 11.6E

1.a. Assume that a sentence **P** is quantificationally false. Then {**P**} is quantificationally inconsistent. It follows from Metatheorem 11.6.1 that every systematic tree for {**P**} closes.

b. Assume that a sentence **P** is quantificationally true. Then ~ **P** is quantificationally false, and {~ **P**} is quantificationally inconsistent. It follows from Metatheorem 11.6.1 that every systematic tree for {~ **P**} closes.

d. Assume that Γ ⊨ **P**. Then on every interpretation on which every member of Γ is true, **P** is true, and ~ **P** is therefore false. So Γ ∪ {~ **P**} is quantificationally inconsistent. It follows from Metatheorem 11.6.1 that every systematic tree for Γ ∪ {~ **P**} closes.

2.a. The lengths are 6, 2, and 6, respectively.

b. Assume that the length of a sentence ~ (**Q** & **R**) is **k**. Then since ~ (**Q** & **R**) contains an occurrence of the tilde and an occurrence of the ampersand that neither **Q** nor **R** contains, the length of **Q** is **k** − 2 or less and the length of **R** is **k** − 2 or less. Hence the length of ~ **Q** is **k** − 1 or less, and the length of ~ **R** is **k** − 1 or less.

d. Assume that the length of a sentence ~ (∀**x**)**Q** is **k**. Then the length of the formula **Q** is **k** − 2. Hence the length of **Q**(**a**/**x**) is **k** − 2, since **Q**(**a**/**x**) differs from **Q** only in containing **a** wherever **Q** contains **x** and neither constants nor variables are counted in computing the length of a formula. Hence the length of ~ **Q**(**a**/**x**) is **k** − 1.

3.a. **P** is of the form **Q** ∨ **R**. Assume that **P** ∈ Γ. Then, by e, either **Q** ∈ Γ, or **R** ∈ Γ. If **Q** ∈ Γ, then **I**(**Q**) = **T**, by the inductive hypothesis. If **R** ∈ Γ, then **I**(**R**) = **T**, by the inductive hypothesis. Either way, it follows that **I**(**Q** ∨ **R**) = **T**.

c. $\mathbf{P}$ is of the form $\mathbf{Q} \supset \mathbf{R}$. Assume that $\mathbf{P} \in \Gamma$. Then, by g, either $\sim \mathbf{Q} \in \Gamma$ or $\mathbf{R} \in \Gamma$. By the inductive hypothesis, then, either $\mathbf{I}(\sim \mathbf{Q}) = \mathbf{T}$ or $\mathbf{I}(\mathbf{R}) = \mathbf{T}$. So either $\mathbf{I}(\mathbf{Q}) = \mathbf{F}$ or $\mathbf{I}(\mathbf{R}) = \mathbf{T}$. Consequently, $\mathbf{I}(\mathbf{Q} \supset \mathbf{R}) = \mathbf{T}$.

f. $\mathbf{P}$ is of the form $\sim (\mathbf{Q} \equiv \mathbf{R})$. Assume that $\mathbf{P} \in \Gamma$. Then, by j, either both $\mathbf{Q} \in \Gamma$ and $\sim \mathbf{R} \in \Gamma$, or both $\sim \mathbf{Q} \in \Gamma$ and $\mathbf{R} \in \Gamma$. In the former case, $\mathbf{I}(\mathbf{Q}) = \mathbf{T}$ and $\mathbf{I}(\sim \mathbf{R}) = \mathbf{T}$, by the inductive hypothesis; so $\mathbf{I}(\mathbf{Q}) = \mathbf{T}$ and $\mathbf{I}(\mathbf{R}) = \mathbf{F}$. In the latter case, $\mathbf{I}(\sim \mathbf{Q}) = \mathbf{T}$ and $\mathbf{I}(\mathbf{R}) = \mathbf{T}$, by the inductive hypothesis; hence $\mathbf{I}(\mathbf{Q}) = \mathbf{F}$ and $\mathbf{I}(\mathbf{R}) = \mathbf{T}$. Either way, it follows that $\mathbf{I}(\mathbf{Q} \equiv \mathbf{R}) = \mathbf{F}$, and so $\mathbf{I}(\sim (\mathbf{Q} \equiv \mathbf{R})) = \mathbf{T}$.

g. $\mathbf{P}$ is of the form $(\exists \mathbf{x})\mathbf{Q}$. Assume that $\mathbf{P} \in \Gamma$. Then, by m, there is some constant $\mathbf{a}$ such that $\mathbf{Q}(\mathbf{a}/\mathbf{x}) \in \Gamma$. By the inductive hypothesis, $\mathbf{I}(\mathbf{Q}(\mathbf{a}/\mathbf{x})) = \mathbf{T}$. By 11.1.5, $\{\mathbf{Q}(\mathbf{a}/\mathbf{x})\} \vdash (\exists \mathbf{x})\mathbf{Q}$. So $\mathbf{I}((\exists \mathbf{x})\mathbf{Q}) = \mathbf{T}$ as well.

5. Clauses 7 and 9. First consider clause 7. Suppose that $\mathbf{Q} \supset \mathbf{R}$ has $\mathbf{k}$ occurrences of logical operators. Then $\mathbf{Q}$ certainly has fewer than $\mathbf{k}$ occurrences of logical operators, and so does $\mathbf{R}$. But, in the proof for case 7, once we assume that $\mathbf{Q} \supset \mathbf{R} \in \Gamma$, we know that $\sim \mathbf{Q}$ or $\mathbf{R}$ is a member of Γ by property g of Hintikka sets. The problem is that we cannot apply the inductive hypothesis to $\sim \mathbf{Q}$ since $\sim \mathbf{Q}$ might contain $\mathbf{k}$ occurrences of logical operators. In the sentence '(Am & Bm) ⊃ Bm', for instance, this happens. The entire sentence has two occurrences of logical operators, but so does the negation of the antecedent '~ (Am & Bm)'. However, it can easily be shown that the *length* of $\sim \mathbf{Q}$ is less than the *length* of $\mathbf{Q} \supset \mathbf{R}$.

Similarly, in the case of clause 9 we know that if $\mathbf{Q} \equiv \mathbf{R} \in \Gamma$, then either both $\mathbf{Q} \in \Gamma$ and $\mathbf{R} \in \Gamma$ or both $\sim \mathbf{Q} \in \Gamma$ and $\sim \mathbf{R} \in \Gamma$. But then we are not guaranteed that either $\sim \mathbf{Q}$ or $\sim \mathbf{R}$ has fewer occurrences of logical operators than does $\mathbf{Q} \equiv \mathbf{R}$. For instance, '~ Am' and '~ Bm' each contain one occurrence of a logical operator, and so does 'Am ≡ Bm'.

6. If ∃D were not included, then we could not be assured that the set of sentences on each open branch of a systematic tree has property m of Hintikka sets. And in the inductive proof that every Hintikka set is quantificationally consistent we made use of this property in steps (12) and (13).

7. Yes, it would. For let us trace those places in our proof of Metatheorem 11.6.1 where we appealed to the rule ~ ∀D. We used it to establish that the set of sentences on an open branch of a systematic tree has property 1 of Hintikka sets, and we appealed to property 1 in step (12) of our inductive proof of 11.6.4. So let us first replace property 1 by the following:

> 1*. If $\sim (\forall \mathbf{x})\mathbf{P} \in \Gamma$, then, for some constant $\mathbf{a}$ that occurs in some sentence in Γ, $\sim \mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma$.

It is then easily established that every open branch of a systematic tree has properties a to k, 1*, and m to n. In our inductive proof of Lemma 11.6.4, change step (12) to the following:

12*. $\mathbf{P}$ is of the form $\sim (\forall \mathbf{x})\mathbf{Q}$. Assume that $\mathbf{P} \in \Gamma$. Then, by 1*, there is some constant **a** such that $\sim \mathbf{Q}(\mathbf{a}/\mathbf{x}) \in \Gamma$. By the inductive hypothesis, $\mathbf{I}(\sim \mathbf{Q}(\mathbf{a}/\mathbf{x})) = \mathbf{T}$, and so $\mathbf{I}(\mathbf{Q}(\mathbf{a}/\mathbf{x})) = \mathbf{F}$. Since $\{(\forall \mathbf{x})\mathbf{Q}\} \vDash \mathbf{Q}(\mathbf{a}/\mathbf{x})$, by 11.1.4, it follows that $\mathbf{I}((\forall \mathbf{x})\mathbf{Q}) = \mathbf{F}$ and $\mathbf{I}(\sim (\forall \mathbf{x})\mathbf{Q}) = \mathbf{T}$.

8. Certain adjustments are obvious if we are to convert the proof of Metatheorem 11.6.1 into a proof that the tree method for *SL* is complete for sentential logic. The tree method for *SL* contains only some of the rules of the tree method for *PL*; hence we have fewer rules to work with. We replace talk of quantificational concepts (consistency and the like) with talk of truth-functional concepts, hence talk of interpretations with talk of truth-value assignments.

A Hintikka set of *SL* will have only properties a to j of Hintikka sets for *PL*. And trees for *SL* are *all* finite, so we have only finite open branches to consider in this case. (Thus Lemma 11.6 would not be used in the proof for *SL*.) Finally, the construction of the characteristic truth-value assignment for a Hintikka set of *SL* requires only clause 2 of the construction of the characteristic interpretation for a Hintikka set of *PL*.

9. We must first show that a set Γ^* that is both maximally consistent in *PD* and ∃-complete has the 14 properties of Hintikka sets. We list those properties here. (And we refer to the 7 properties a to g of sets that are both maximally consistent in *PD* and ∃-complete as ‘M(a)’, ‘M(b)’, . . . , ‘M(g)’.)

a. For any atomic sentence $\mathbf{P}$, not both $\mathbf{P}$ and $\sim \mathbf{P}$ are members of Γ^*.

Proof: This follows immediately from property M(a) of Γ^*.

b. If $\sim \sim \mathbf{P}$ is a member of Γ^*, then $\mathbf{P}$ is a member of Γ^*.

Proof: If $\sim \sim \mathbf{P} \in \Gamma^*$, then $\sim \mathbf{P} \notin \Gamma^*$, by M(a), and $\mathbf{P} \in \Gamma^*$, by M(a).

c. If $\mathbf{P} \;\&\; \mathbf{Q} \in \Gamma^*$, then $\mathbf{P} \in \Gamma^*$ and $\mathbf{Q} \in \Gamma^*$.

Proof: This follows from property M(b) of Γ^*.

d. If $\sim (\mathbf{P} \;\&\; \mathbf{Q}) \in \Gamma^*$, then either $\sim \mathbf{P} \in \Gamma^*$ or $\sim \mathbf{Q} \in \Gamma^*$.

Proof: If $\sim (\mathbf{P} \;\&\; \mathbf{Q}) \in \Gamma^*$, then $\mathbf{P} \;\&\; \mathbf{Q} \notin \Gamma^*$, by M(a). By M(b), either $\mathbf{P} \notin \Gamma^*$ or $\mathbf{Q} \notin \Gamma^*$. By M(a), either $\sim \mathbf{P} \in \Gamma^*$ or $\sim \mathbf{Q} \in \Gamma^*$.

e. to j. are established similarly.

k. If $(\forall \mathbf{x})\mathbf{P} \in \Gamma$, then at least one substitution instance of $(\forall \mathbf{x})\mathbf{P}$ is a member of Γ and for every constant **a** that occurs in some sentence of Γ, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma$.

Proof: This follows from property M(f) of Γ^*.

l. If $\sim (\forall \mathbf{x})\mathbf{P} \in \Gamma^*$, then $(\exists \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$.

Proof: If $\sim (\forall \mathbf{x})\mathbf{P} \in \Gamma^*$, then $(\forall \mathbf{x})\mathbf{P} \notin \Gamma^*$, by M(a). Then, for some constant $\mathbf{a}$, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \notin \Gamma^*$, by M(f). Then $\sim \mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma^*$, by M(a). So $(\exists \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$, by M(g).

m. If $(\exists \mathbf{x})\mathbf{P} \in \Gamma^*$, then, for at least one constant $\mathbf{a}$, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma^*$.

Proof: This follows from property M(g) of Γ^*.

n. If $\sim (\exists \mathbf{x})\mathbf{P} \in \Gamma^*$, then $(\forall \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$.

Proof: If $\sim (\exists \mathbf{x})\mathbf{P} \in \Gamma^*$, then $(\exists \mathbf{x})\mathbf{P} \notin \Gamma^*$, by M(a). Then, for every constant $\mathbf{a}$, $\mathbf{P}(\mathbf{a}/\mathbf{x}) \notin \Gamma^*$, by M(g). So, for every constant $\mathbf{a}$, $\sim \mathbf{P}(\mathbf{a}/\mathbf{x}) \in \Gamma^*$, by M(a). And $(\forall \mathbf{x}) \sim \mathbf{P} \in \Gamma^*$, by M(f).

Second, that every Hintikka set is ∃-complete follows from property m of Hintikka sets.

Third, we show that some Hintikka sets are *not* maximally consistent in *PD*. Here is an example of such a set:

$\{(\forall x)Fx,\ (\exists y)Fy,\ Fa\}$

It is easily verified that this set is a Hintikka set. And the set is of course consistent in *PD*. But this set is *not* such that the addition to the set of any sentence that is not already a member will create an inconsistent set. For instance, the sentence ‘Fb’ may be added, and the resulting set is also consistent in *PD*:

$\{(\forall x)Fx,\ (\exists y)Fy,\ Fa,\ Fb\}$

Hence the set is not maximally consistent in *PD*.