4 Relative Modal Logic – System K

A. System K

1. Absolute versus Relative Modalities

A modal statement is one whose main connective is a modal operator (□,◇), or is the negation of such a statement.[1]

In Absolute Modal Logic, which is formalized by System L, every modal statement is *absolute*, in the following sense: if it is true at *any* reference point (index, possible world), then it is true at *every* reference point.[2] This is reflected in the associated absolute (Leibnizian) world-theory, in which the notion of possibility is absolute, or non-relational. In particular, in absolute world-theory, a world is possible *simpliciter*, not in *relation* to something else.

In contrast to absolute world-theory, there is relative world-theory, according to which the notion of possibility is relational; in particular, a world is possible or impossible, not *simpliciter*, but *relative to* a given world. Moreover, a proposition is necessary at a given world i if and only if it is true at every world that is possible relative to i. This can formalized as follows.

$$[\Box\mathbb{S} / i] \leftrightarrow \forall j\{ \mathcal{R}ij \rightarrow [\mathbb{S} / j] \}$$

Here,

$\mathcal{R}ij$

reads:

j is possible *relative to* i.

See Part B for a more detailed account of relative world-theory.

Associated with relative world-theory there is Relative Modal Logic. The key feature of Relative Modal Logic is that a modal statement can be true at a given reference point without automatically being true at every reference point. This enables one to construct logical systems in which iterated modalities are not redundant. On the other hand, whereas there is essentially just one absolute modal logic, there are infinitely-many non-equivalent relative modal logics.

We will consider several of these, the first one being system K.

2. Index Points in Relative Modal Logic

In Absolute Modal Logic, index points are syntactically represented by decimal Arabic numerals. By contrast, in Relative Modal Logic, index points are syntactically represented by finite *sequences* of decimal numerals. There are official restrictions on what sequences are permitted in a given derivation, although these restrictions are primarily intended to simplify and sanitize derivation-construction. The official technical definition is given later. One basically learns to apply the official definition by seeing examples.

Let us do an example. The following is an example of an admissible set of indices.

0, 01, 02, 013, 024, 025, 0136, 0137

---

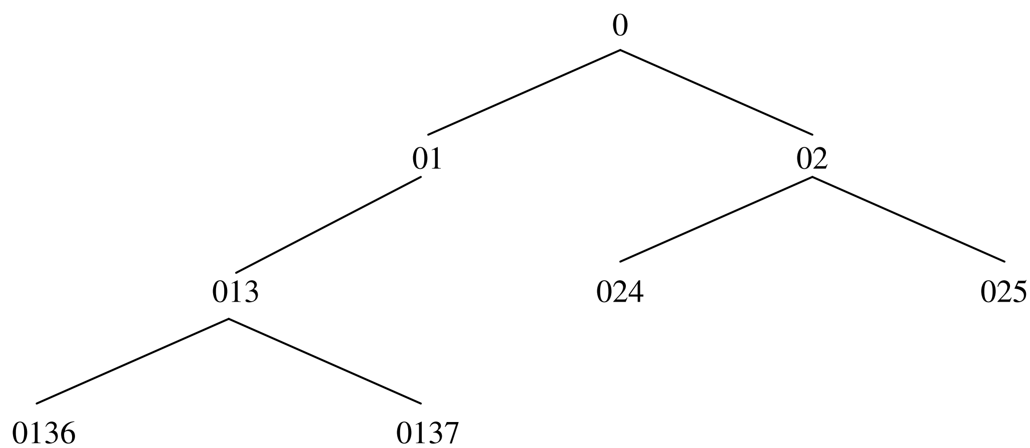
[1] Henceforth, we largely ignore the strict conditional and strict biconditional, treating them as purely derived. Also, in this connection, see Section 9.

[2] Recall that the contradiction formula ※ is absolute in exactly the same sense.

These follow the "rules" [see below]; for example, every string starts with 0, every string has a unique last character, and no digit is skipped.

By way of understanding exactly how indices are inter-related, it is useful to regard index points as compound names, in particular as patronymic names (or matronymic, if you prefer). In particular, each compound name consists of a unique "given" name (its unique terminal numeral) affixed to the name of its immediate ancestor. The immediate ancestor, in turn, is named in precisely the same manner – a unique given name is affixed to the name of its immediate ancestor. In this way, given an index, we can trace its ancestry all the way back to the original index, 0.[3]

The patronymic system can be further clarified by graphing the associated family tree. For example, the index system above can be graphed as follows.



In this tree, we see the following. 0 has two immediate descendants, 01 and 02. 01 has just one immediate descendant, 013, which has two immediate descendants, 0136 and 0137. 02 has two immediate descendants, 024 and 025. The terminal nodes – 0136, 0137, 024, 025 – have no descendants.

Note carefully that, modal-logically speaking, descendancy *is* relative possibility; for example, 01 and 02 are both possible relative to 0, and 0 is not possible relative to anything.

3. A Potential Notational Problem

We do have a potential confound in our notation. In particular, index names are strings of decimal numerals, and decimal numerals are strings of digits. So, if we have 11 indices, we must use the numeral ‘10’, in which case we might have the string ‘010’. This might be confused with the sequence. But, if we examine the rules of indices, we see that zero-one-zero is disallowed, although zero-ten is allowed. The reader is invited to consider other strings of digits to see if any genuine ambiguity arises.

In any event, be assured that, although the index-naming scheme may produce theoretical problems for *general* derivations, no problem arises for any *actual* derivation considered in this work.

[3] Note also that we are speaking about compound *names*, not compound *individuals*. Although the names are compound, the objects they name are (presumably) simple.

4. The Official Technical Definition

As mentioned earlier, one can learn the rules of indexing pretty much the same way we learn all language – by immersion. Nevertheless, for the sake of logical thoroughness, and in case we have to teach it to a computer, we offer the following official definition.

(d1) Let Σ be a set of sequences of numerals. Then Σ is an admissible set of indices for relative modal logic if and only if it satisfies the following restrictions.

(1) $\Sigma \neq \varnothing$;
(2) for every $\sigma \in \Sigma$, first(σ) = 0;
(3) for every $\sigma_1, \sigma_2 \in \Sigma$, if last($\sigma_1$) = last($\sigma_2$), then $\sigma_1 = \sigma_2$;
(4) for every $\sigma \in \Sigma$, if $\sigma' \leq \sigma$, then $\sigma' \in \Sigma$;
(5) if $\sigma \in \Sigma$, and last(σ)=n, and m<n, then $\exists\sigma'\in\Sigma$: last(σ')=m

In other words: (1) Σ is non-empty; (2) every σ in Σ begins with 0; (3) every σ has a unique terminal (last) element; (4) Σ is closed with respect to the predecessor relation [e.g., if 0136 is in Σ, then so are 0, 01, and 013]; (5) numerals are not skipped.

5. The Basic Rules of System K

With the new concept of index in mind, let us examine the rules of System K.

In the following rules, whereas the variables 'i' and 'j' range over indices (i.e., numerical sequences), the variables 'm' and 'n' range over numerals. The expression 'i+m' denotes the sequence that results when *numeral* m is appended to *sequence* i [for example, 03+7=037].

□O		◇I	
□Φ	/i	Φ	/i+m (old)
———		———	
Φ	/i+m (old)	◇Φ	/i

ND (□D)		◇O	
~~SHOW~~: □Φ	/i	◇Φ	/i
| ~~SHOW~~: Φ	/i+n (new)	———	
| |		Φ	/i+n (new)

6. Indices – Old and New

Notice that the rules involve the concepts *old* and *new*. We have already seen the term 'new' in System L, which is borrowed from elementary quantifier logic. The following is our official definitions of *old* and *new*, which is followed by a new general policy about applying rules of derivation.[4]

[4] When we move from ordinary quantifier logic to universally-free quantifier logic (UFQL),[4] we have to make similar adjustments in our rules, including the official definition of 'new'. UFQL allows for the possibility that the domain is empty!

> An index counts as ***old*** precisely when it occurs in at least one line that is neither boxed nor cancelled; otherwise, it counts as ***new***.

> Unless a rule specifically requires that an index is new (e.g., □D/ND), it is understood that all the relevant indices are old.

This policy is officially required in order to prohibit "sneaky" derivations of invalid argument forms. On the other hand, it is almost always automatically satisfied in "straightforwardly" constructed derivations. See the appendix for an examination of the technical problems that arise in System K.

Notice that, since every derivation has an initial show-line indexed by 0, and this line never gets boxed, the index 0 counts as old throughout the derivation, at least up to the point when this line is cancelled. Notice also that, given the definition of admissible index, in order for index i+n to be new, the numeral n must be new, which is to say that n does not occur in any old index.

7. Simple Examples of Derivations in System K

In order to illustrate the rules, let us examine some examples.

Example 1: □(P → Q) ; □P / □Q

(1)	□(P → Q)	/0	Pr	
(2)	□P	/0	Pr	
(3)	~~SHOW~~: □Q	/0	ND	
(4)	~~SHOW~~: Q	/01	DD	new
(5)	│ P→Q	/01	1,□O	old
(6)	│ P	/01	2,□O	old
(7)	│ Q	/01	5,6,SL	

In this derivation, two indices/worlds are considered – 0 and 01; moreover, 01 is an immediate descendant of 0. Both worlds are *generically* possible, but world 01 is furthermore *possible relative to* world 0.

Example 2: □(P → Q) ; ◇P / ◇Q

(1)	□(P → Q)	/0	Pr	
(2)	◇P	/0	Pr	
(3)	~~SHOW~~: ◇Q	/0	DD	
(4)	│ P	/01	2,◇O	new
(5)	│ P → Q	/01	1,□O	old
(6)	│ Q	/01	4,5,SL	
(7)	│ ◇Q	/0	6,◇I	01 old

Example 3: □(□P → □(P → Q)) ; □□P / □□Q

This means that the argument form [∀xFx / ∃xFx] is invalid in UFQL. When we implement UFQL within a natural deduction system, we must adjust ∀O so that it can only be applied to *old* constants.

(1)	□(□P → □(P → Q))	/0	Pr	
(2)	□□P	/0	Pr	
(3)	~~SHOW~~: □□Q	/0	ND	
(4)	~~SHOW~~: □Q	/01	ND	new
(5)	~~SHOW~~: Q	/012	DD	
(6)	│ □P → □(P → Q)	/01	1,□O	old
(7)	│ □P	/01	2,□O	old
(8)	│ □(P → Q)	/01	6,7,SL	
(9)	│ P	/012	7,□O	old
(10)	│ P → Q	/012	8,□O	old
(11)	│ Q	/012	9,10,SL	

8. Modal-Negation Rules

We next consider the official modal negation rules for System K, which are the same as for System L.

MN	
~□𝒜 ═════ ◇~𝒜	~◇𝒜 ═════ □~𝒜

Notice that the indices are suppressed. It is understood that input-index and the output-index are the *same*, and it is understood that the index is *old*. (See Appendix)

9. Short-Cut Modal-Negation Rules

As in System L, System K also has short-cut modal negation rules. As before, ~□O is simply a combination of MN and ◇O, and ~◇O is simply a combination of MN and □O.

~□O = MN+◇O	~◇O = MN+□O
~□𝒜 /*i* ――― ~𝒜 /*i*+*n* (new)	~◇𝒜 /*i* ――― ~𝒜 /*i*+*m* (old)

10. Examples of Derivations that Employ Modal-Negation

Example 4: □(P → Q) ; ~□Q / ~□P

(1)	□(P → Q)	/0	Pr	
(2)	~□Q	/0	Pr	
(3)	~~SHOW~~: ~□P	/0	ID	
(4)	│ □P	/0	As	
(5)	│ ~~SHOW~~: ※	/*	DD	
(6)	│ │ ~Q	/01	2,~□O	new
(7)	│ │ P→Q	/01	1,□O	old
(8)	│ │ P	/01	4,□O	old

Example 5: □(P → Q) ; ~◇Q / ~◇P

(1)	□(P → Q)	/0	Pr	
(2)	~◇Q	/0	Pr	
(3)	~~SHOW~~: ~◇P	/0	ID	
(4)	│ ◇P	/0	As	
(5)	│ ~~SHOW~~: ※	/*	DD	
(6)	│ │ P	/01	4,◇O	new
(7)	│ │ P→Q	/01	1,□O	old
(8)	│ │ ~Q	/01	2,~◇O	old
(9)	│ │ ※	/*	6,7,8,SL	

11. The Strict-Conditional and Strict-Biconditional in Relative Modal Logic

Recall that, in Absolute Modal Logic, the strict-conditional (⥽) is defined as a necessary conditional, and the strict-biconditional (=) is defined as a necessary biconditional. We could do the same thing in Relative Modal Logic.

However, we choose to restrict the strict-conditional and the strict-biconditional to Absolute Modal Logic. We do not use them in System K. They do re-appear later – when we resurrect the absolute modal operators.

12. Some Prominent Non-Derivations

Completed derivations are instructive, but so are non-completed derivations. Consider the following zero-premise argument forms.

(1) / □P → P
(2) / □P → ◇P
(3) / □P → □□P
(4) / ◇□P → P
(5) / ◇□P → □P
(6) / ◇□P → □◇P

Each of these is provable in System L, which means that each formula is a logical truth according to System L. What about System K? Well, not even one of them can be proved in System K! The following is the best we can do.

#1

(1)	SHOW: □P → P	/0	CD
(2)	□P	/0	As
(3)	SHOW: P	/0	??

In System K, we cannot go from □P at 0 to P at 0, because 0 is not an immediate descendent of itself. From 0, we can go to 01, to 02, to 03, but we cannot go to 0!

The following is an interesting, but nevertheless <u>illegitimate derivation</u>; can you figure out which rule it violates and why?

#2

(1)	~~SHOW~~: $\Box P \rightarrow \Diamond P$	/0	CD	??
(2)	$\Box P$	/0	As	??
(3)	~~SHOW~~: $\Diamond P$	/0	DD	??
(4)	~~SHOW~~: P	/01	DD	??
(5)	P	/01	2,$\Box$O	??
(6)	$\Diamond P$	/0	4,$\Diamond$I	??

#3

(1)	SHOW: $\Box P \rightarrow \Box\Box P$	/0	CD
(2)	$\Box P$	/0	As
(3)	SHOW: $\Box\Box P$	/0	ND
(4)	SHOW: $\Box P$	/01	ND
(5)	SHOW: P	/012	??
(6)	P	/01	2,$\Box$O

Notice that, in System K, we cannot go from $\Box\mathcal{A}$ at 0 to $\mathcal{A}$ at 012. Remember, in System K, $\Box$O allows one to go from $\Box$P at an index i to P at any immediate descendant of i. Although 012 is an immediate descendent of 01, which is an immediate descendent of 0, 012 is not an immediate descendent of 0.

#4

(1)	SHOW: $\Diamond\Box P \rightarrow P$	/0	CD
(2)	$\Diamond\Box P$	/0	As
(3)	SHOW: P	/0	
(4)	$\Box P$	/01	2,$\Diamond$O

We cannot finish this derivation, for although 01 is an immediate descendent of 0, 0 is not an immediate descendent of 01.

#5

(1)	SHOW: $\Diamond\Box P \rightarrow \Box P$	/0	CD
(2)	$\Diamond\Box P$	/0	As
(3)	SHOW: $\Box P$	/0	ND
(4)	SHOW: P	/01	
(5)	$\Box P$	/02	2,$\Diamond$O

We have $\Box$P at 02; we want P at 01, but we have no way of getting from 02 to 01; they are both immediate descendants of 0, but neither is an immediate descendent of the other (no incest!)

#6

(1)	SHOW: $\Diamond\Box P \rightarrow \Box\Diamond P$	/0	CD
(2)	$\Diamond\Box P$	/0	As
(3)	SHOW: $\Box\Diamond P$	/0	ND
(4)	SHOW: $\Diamond P$	/01	
(5)	$\Box P$	/02	

If we knew that index 01 and index 02, both immediate descendants of 0, had a common immediate descendent (incest!), then we could finish, but we have no such information.

13. Why is System K such a Weakling?

At this point, the reader may be wondering why we have created a system without any of the above theses. There are two, inter-related, reasons.

Reason #1:

We are trying to devise a system of modal logic that is sufficiently general to encompass as many specific modalities as possible. We want a logic of '☐' that captures *all* box-modalities. By way of illustration, let us do the simplest example.

(1) $\Box P \rightarrow P$

Obviously if we read '☐P' as

it is *necessary* that P,

then (1) will be a valid formula. Similarly, it is valid, if we read '☐P' as

it is *regrettable* that P
or as: it is *known* that P
or as: it is *true* that P

On the other hand, if we read '☐P' as

it *ought* to be the case that P
or as it is generally *believed* that P
or as it has frequently been *said* that

then the formula $\Box P \rightarrow P$ should most certainly not be valid. The reader is invited to consider the other formulas above in connection with different readings of '☐'.

Reason #2:

System K has the following trait. Suppose modal system Σ is characterized by a Kripke-style world theory [see Appendix 2]. Then every thesis of K is a thesis of Σ, and conversely, every thesis of Σ is a thesis of K. In other words, System K is what all Kripke-style modal logics have in common. That is why we call it System K.

Another way to describe this is as follows. System K is the smallest *normal* modal logic. What is the strongest normal modal logic? Practically speaking, it is System L.[5] So all normal modal logics are at least as strong as K, but no stronger than L.

[5] Technically speaking, L is the strongest normal modal logic that does not have a finite characteristic matrix. For example, we can obtain a stronger system than L by adding the thesis $P \rightarrow \Box P$, but the resulting system is characterized by a two-element matrix, which trivializes the resulting modal logic.

14. Appendix: Issues Concerning *Old* and *New*

1. A Problem with *Old* and *New*

The reason we have to redefine old/new in the context of System K [and in universally-free logic] is because of the way certain other rules are written. The first problem concerns show-lines, whose structural-rule is officially written as follows.

> At any point in a derivation, one can write down <u>any</u> show-line, with <u>any</u> index.

This is a critically important rule, but it makes System K unexpectedly complicated. Consider the following argument form, which is <u>not valid</u> in System K – as it is historically given.

□P / ◇P

Unfortunately, if we are not careful about officially stating our rules, we will allow derivations that prove this argument to be valid. This is not good!

In the following, we consider various counterfeit derivations of the argument form ⟨□P/◇P⟩, and we show how our rules bust them.

#1

(1)	□P	/0	Pr	
(2)	~~SHOW~~: ◇P	/0	DD	
(3)	│ ~~SHOW~~: P	/01	DD	
(4)	│ │ P	/01	1,□O	
(5)	│ ◇P	/0	3,◇I	※※※

The culprit is line (5); which claims to derive from line (3) by ◇I. But according to ◇I for System K, the index of the input line must be <u>old</u> at the time of application, which is at line 5. The question then is whether index 01 is old. Now, if all we require is that 01 occur in an earlier available line, then it does, because 01 occurs in line (3), which is surely both earlier and available. On the other hand, if we use our new definition of 'old', since the earlier available line is a cancelled show-line, the index 01 does not count as old. Thus, the new definition of *new/old* renders this derivation inadmissible (which is good, since it is invalid in System K!).

#2

(1)	□P	/0	Pr	
(2)	~~SHOW~~: ◇P	/0	DD	
(3)	│ ~~SHOW~~: P	/01	DD	
(4)	│ │ P	/01	1□O	
(5)	│ P	/01	1,□O	※※※
(6)	│ ◇P	/0	4,◇I	

In this case, the application of ◇I at line-6 is ok, since the index 01 occurs in an available earlier line – line-5 – which is not a cancelled show-line. But line-5 is not ok. The problem is that it allegedly follows from line-1 by □O. But □O requires that the index be old at the time of application. But it isn't, because the only unboxed occurrence of 01 is in line-3, and line-3 is a cancelled show-line.

#3

(1)	□P	/0	Pr	
(2)	~~SHOW~~: ◇P	/0	DD	
(3)	│ ~~SHOW~~: P → P	/01	CD	
(4)	│ │ P	/01	As	
(5)	│ │ ~~SHOW~~: P	/01	DD	
(6)	│ │ │ P	/01	4,R	
(7)	│ P	/01	1,□O	⨳⨳⨳
(8)	│ ◇P	/0	4,◇I	

Notice that line-3 is legitimate, even if it seems a bit off-the-wall. Remember – one is entitled to write down any show-line, with any index, at any time in a derivation. So far, so good. The culprit in this derivation is line-7. □O requires an *old* index, but the only available earlier occurrence of 01 is in line-3, which is a cancelled show-line.

2. Yet Another Problem with *Old/New*

We are not out of the woods yet. For, consider the following counterfeit derivation.

#4

(1)	□P	/0	Pr	
(2)	~~SHOW~~: ◇P	/0	DD	
(3)	│ P → P	/01	SL	⨳⨳⨳
(4)	│ P	/01	1,□O	
(5)	│ ◇P	/0	4,◇I	

Here, since P→P is a tautology, it follows from "nothing" by the rule SL. This is the Tautology-Rule, which *in non-modal logic* may be stated as follows.

(T) at any point in a derivation, one can write down <u>any</u> tautology.

The natural adaptation to indexed logic is the following.

✕ at any point in a derivation, one can write down <u>any</u> tautology, with <u>any</u> index.

Unfortunately, this gets us into serious trouble. So we adjust it as follows.[6]

> At any point in a derivation, one can write down <u>any</u> tautology,
> provided (and *only* provided) the index is **old**.

Now, we see where example #4 goes astray – on line-3. In particular, line-3 is dis-allowed, since the index 01 is not old.

Unfortunately, we are *still* not out of the woods. For consider the following counterfeit derivation.

[6] Note also that, since the tautology-rule is a zero-place rule, we must also adjust the corresponding structural-rule about zero-place rules, as follows.

At any point, in any derivation, one can apply <u>any</u> zero-place rule, provided (and *only* provided) the index is **old**.

#5

(1)	$\Box$P	/0	Pr	
(2)	~~SHOW~~: $\Diamond$P	/0	DD	
(3)	~~SHOW~~: Q $\rightarrow$ Q	/01	CD	
(4)	Q	/01	As	
(5)	~~SHOW~~: Q	/01	DD	
(6)	Q	/01	4,R	
(7)	Q $\rightarrow$ Q	/01	3,R	※※※
(8)	P	/01	1,$\Box$O	
(9)	$\Diamond$P	/0	4,$\Diamond$I	

Here, R is the rule of repetition, originally from SL, which seems pretty innocuous. Notice that line-8 is ok, since its index 01 does occur in a previous available line that is not a cancelled show-line. The problem, intuitively, is that line-7 merely restates line-3, which *is* a cancelled show-line.

So, we have to adjust the repetition rule when it is adapted to indexed logic. In particular, rule R can be written as follows.

R		
	$\mathcal{A}$	/i (old)
	——	
	$\mathcal{A}$	/i

But this is not the only SL-rule we have to adjust. Consider double-negation (DN) as used in the following counterfeit derivation.

#6

(1)	$\Box$P	/0	Pr	
(2)	~~SHOW~~: $\Diamond$P	/0	DD	
(3)	~~SHOW~~: Q $\rightarrow$ Q	/01	CD	
(4)	Q	/01	As	
(5)	~~SHOW~~: Q	/01	DD	
(6)	Q	/01	4,R	
(7)	$\sim\sim$(Q $\rightarrow$ Q)	/01	3,DN	※※※
(8)	P	/01	1,$\Box$O	
(9)	$\Diamond$P	/0	4,$\Diamond$I	

Once again, the faulty line involves an application of an SL rule, and once again the correction involves a proper adjustment of the SL rule, given as follows.

DN		
	$\mathcal{A}$	/i (old)
	——	
	$\sim\sim\mathcal{A}$	/i

These sorts of examples can be repeated indefinitely. What we need is a general policy; that is given as follows.

Policy

Unless a rule specifically requires that an index is new (e.g., □D), it is understood that all the indices are old.

3. Gratuitous Show-Lines

Recall that the general structural-rule about show-lines is quite simple.

(S) At any point in a derivation, one can write down <u>any</u> show-line, with <u>any</u> index.

Now, a *gratuitous* show-line is one that seems to come out of thin air; it is not dictated by a particular rule or strategy. In intro logic, the wedge-out strategy employs a gratuitous show-line.

Although, one is entitled to write down gratuitous show-lines, and they can used very creatively, there is an important rule of thumb to follow.

A gratuitous show-line should be written down *only if* its index is old.

The reason is that, although we are entitled to write down any show-line we wish, if a show-line is not dictated by a strategy, in which case it is gratuitous, that show-line will be completely useless unless its index is old.

B. Kripkean World Theory

1. Introduction

In an earlier chapter, we examined Leibnizian World Theory, which is intimately related to Modal System L. In this appendix, we examine Kripkean World Theory, which is similarly related to Modal System K.

Recall the central thesis of Leibnizian World Theory.

a proposition is necessary if and only if it is true in every possible world.

We can formalize this in various ways, including the original strictly first-order formulation,

$$\forall x\{Px \rightarrow [Nx \leftrightarrow \forall y(Wy \rightarrow Txy)]\},$$

and the more tractable, but not *strictly* first-order, formulation.

$$[\Box\mathcal{A} \,/\, i] \leftrightarrow \forall j[\mathcal{A} \,/\, j].$$

Also, recall the associated "dual" principle.

$$[\Diamond\mathcal{A} \,/\, i] \leftrightarrow \exists j[\mathcal{A} \,/\, j]$$

The latter two formulas can be made somewhat more explicit by including a possible-world predicate, as follows.

$$[\Box\mathcal{A} \,/\, i] \leftrightarrow \forall j\{Pj \rightarrow [\mathcal{A} \,/\, j]\}$$

$$[\Diamond\mathcal{A} \,/\, i] \leftrightarrow \exists j\{Pj \;\&\; [\mathcal{A} \,/\, j]\}$$

Here 'Pj' means 'j is a possible world', or simply, 'j is possible'.

In this scheme, possibility is *absolute* (non-relational): a world is possible *simpliciter*, not in relation to something else. This is grammatically represented by the fact that 'P' is a one-place predicate.

The move from absolute modality to relative modality involves replacing the non-relational notion of possibility by a relational notion. This was originally proposed by Saul Kripke; so we refer to this brand of world theory as 'Kripkean World Theory'.

Now, grammatically, the move from absolute modality to relative modality is accomplished by replacing the one-place predicate 'j is possible' by a two-place predicate 'j is possible relative to i'. This yields a corresponding change in the characterization of modality.

$\Box\mathcal{A}$ (is true) at world i iff $\mathcal{A}$ (is true) at every world possible *relative to* i. $\Diamond\mathcal{A}$ (is true) at world i iff $\mathcal{A}$ (is true) at some world possible *relative to* i.

It is customary to use the letter '$\mathcal{R}$' for **r**elative possibility, and it is customary to read '$\mathcal{R}ij$' as 'j is possible relative to i'. This yields the following reformulation.

$[\Box\mathcal{A} / i] \leftrightarrow \forall j\{\mathcal{R}ij \rightarrow [\mathcal{A} / j]\}$
$[\Diamond\mathcal{A} / i] \leftrightarrow \exists j\{\mathcal{R}ij \ \& \ [\mathcal{A} / j]\}$

In this connection it is sometimes helpful to read '$\mathcal{R}ij$' as 'i *reaches* j', or 'i *sees* j'. Roughly, these say that i makes $\Box\mathcal{A}$ true iff every world within i's reach makes $\mathcal{A}$ true, and i makes $\Diamond\mathcal{A}$ true iff some world within i's reach makes $\mathcal{A}$ true.

The relation expressed by '$\mathcal{R}$' is also often called the *accessibility relation*. In this case, we can say that i makes $\Box\mathcal{A}$ true iff every world *accessible* to i makes $\mathcal{A}$ true.

Finally, since '$\mathcal{R}$' is a *special* functor (like '=' and '/'), we will emphasize this fact by

(1) assigning it a special symbol (logogram) – '$\prec$'.
(2) writing it in its natural position, which is infix.

This yields following reformulation.

(1) $[\Box\mathcal{A} / i] \leftrightarrow \forall j\{ i{\prec}j \rightarrow [\mathcal{A} / j] \}$
(2) $[\Diamond\mathcal{A} / i] \leftrightarrow \exists j\{ i{\prec}j \ \& \ [\mathcal{A} / j] \}$

2. The Official Language of Kripkean World Theory

The underlying syntax of Kripkean World Theory is obtained by adding the predicate '$\prec$' to our the language of Leibnizian World Theory, with the following rule of formation.

(f) if τ_1 and τ_2 are singular terms, then $[\tau_1{\prec}\tau_2]$ is a formula.

Notice that, since '$\prec$' is written in infix notation, outer parentheses are officially required; however, since there is almost never any danger of incorrectly parsing expressions involving '$\prec$', we generally drop the outer parentheses.

3. Theses of Kripkean World Theory

As with Leibnizian World Theory [WT(L)], Kripkean World Theory [WT(K)], has axioms and theorems. For the most part, the axioms of WT(K) are exactly the same as the axioms of WT(L). The only difference concerns the axioms that pertain to the modal operators. The new axioms are presented as follows.

kwt($\Box$) $[\Box\mathcal{A} / i] \quad \leftrightarrow \quad \forall j \{ i{\prec}j \rightarrow [\mathcal{A} / j] \}$
kwt($\Diamond$) $[\Diamond\mathcal{A} / i] \quad \leftrightarrow \quad \exists j \{ i{\prec}j \ \& \ [\mathcal{A} / j] \}$

As before, '$\mathcal{A}$' and '$\mathcal{B}$' schematically represent arbitrary formulas, and the index variable 'i' is understood to be universally quantified.

In one sense, WT(K) is *better* than WT(L): given the presence of the $\prec$-predicate, we can make finer distinctions in WT(K) than we can in WT(L). Another way to express this: WT(L) is a *special case* of WT(K), the special case in which we add to WT(K) the following additional axiom.

(aL) $\forall i \forall j [i{\prec}j]$

This just says that every world sees every world. Being a (generically) possible world and being an accessible world are the same.

On the other hand, in one very important sense, WT(K) is weaker than WT(L), as given in the following two-part metatheorem.

(a) If $\mathbb{F}$ is a thesis of WT(K), and $\mathbb{F}$ does not contain any occurrence of the predicate '$\prec$', then $\mathbb{F}$ is also a thesis of WT(L).

(b) There is at least one formula $\mathbb{F}$ such that: $\mathbb{F}$ is a thesis of WT(L), and $\mathbb{F}$ contains no occurrence of the predicate '$\prec$', but $\mathbb{F}$ is not a thesis of WT(K).

For example, the following are all theorems of WT(K), and hence WT(L).

(t1) $\Box(\mathcal{A}\to\mathcal{B})\to(\Box\mathcal{A}\to\Box\mathcal{B})$
(t2) $\Box(\mathcal{A}\to\mathcal{B})\to(\Diamond\mathcal{A}\to\Diamond\mathcal{B})$
(t3) $\Box(\mathcal{A}\&\mathcal{B}) \leftrightarrow (\Box\mathcal{A} \;\&\; \Box\mathcal{B})$
(t4) $\Diamond(\mathcal{A}\vee\mathcal{B}) \leftrightarrow (\Diamond\mathcal{A} \vee \Diamond\mathcal{B})$
(t5) $\sim\Diamond\mathcal{A} \leftrightarrow \Box\sim\mathcal{A}$

By contrast, the following are theorems of WT(L), but not WT(K).

(L1) $\Box P \to P$
(L2) $\Box P \to \Box\Box P$
(L3) $\Diamond\Box P \to \Box P$
(L4) $\Diamond\Box P \to \Box\Diamond P$
(L5) $\Diamond\Box P \to P$

By way of illustrating (but not proving) part (a), we examine proofs of two theorems we proved earlier in WT(L), but this time we prove them in WT(K).

Note: as before, we cite each axiom by reference to the operator it pertains to (except the "null" axiom). In case of the axioms pertaining to the modal operators, we use the citations 'kwt(□)' and 'kwt(◇)'.

4. Examples of Derivations in WT(K)

Example 1

(1)	~~SHOW~~: $\Box(P\to Q)\to(\Box P\to\Box Q)$	wt(0)
(2)	~~SHOW~~: $\{\Box(P\to Q)\to(\Box P\to\Box Q)\}/0$	wt(→)
(3)	~~SHOW~~: $[\Box(P\to Q) / 0] \to [(\Box P\to\Box Q) / 0]$	CD
(4)	│ $\Box(P\to Q) / 0$	As
(5)	│ ~~SHOW~~: $(\Box P\to\Box Q) / 0$	wt(→)
(6)	│ ~~SHOW~~: $[\Box P / 0] \to [\Box Q / 0]$	CD
(7)	│ │ $\Box P / 0$	As
(8)	│ │ ~~SHOW~~: $\Box Q / 0$	kwt(□)
(9)	│ │ ~~SHOW~~: $\forall i\{0\prec i \to [Q / i]\}$	UCD
(10)	│ │ │ $0\prec 1$	As
(11)	│ │ │ ~~SHOW~~: $Q / 1$	DD
(12)	│ │ │ │ $\forall i\{0\prec i \to [(P\to Q) / i]\}$	4,kwt(□)
(13)	│ │ │ │ $(P\to Q) / 1$	10,12,QL
(14)	│ │ │ │ $[P / 1] \to [Q / 1]$	13,wt(→)
(15)	│ │ │ │ $\forall i\{0\prec i \to [P / i]\}$	7,kwt(□)
(16)	│ │ │ │ $P / 1$	10,15,QL
(17)	│ │ │ │ $Q / 1$	14,16,SL

Example 2

(1)	~~SHOW~~: $\sim\Diamond P \rightarrow \Box\sim P$	wt(0)
(2)	~~SHOW~~: $(\sim\Diamond P \rightarrow \Box\sim P) / 0$	wt(→)
(3)	~~SHOW~~: $[\sim\Diamond P / 0] \rightarrow [\Box\sim P / 0]$	CD
(4)	$\sim\Diamond P / 0$	As
(5)	~~SHOW~~: $\Box\sim P / 0$	wt(□)
(6)	~~SHOW~~: $\forall i\{0 \prec i \rightarrow [\sim P / i]\}$	UCD
(7)	$0 \prec 1$	As
(8)	~~SHOW~~: $\sim P / 1$	wt(~)
(9)	~~SHOW~~: $\sim[P / 1]$	ID
(10)	$P / 1$	As
(11)	~~SHOW~~: ※	12,14,SL
(12)	$\sim[\Diamond P / 0]$	4,wt(~)
(13)	$\exists x\{0 \prec x \;\&\; [P / x]\}$	7,10,QL
(14)	$\Diamond P / 0$	13,wt(◇)

5. Examples of Non-Derivations

Concerning the negative half of the above metatheorem, let us examine *attempted* proofs of (L1) and (L2).

Example 1

(1)	SHOW: $\Box P \rightarrow P$	CD
(2)	$\Box P$	As
(3)	SHOW: P	wt(0)
(4)	SHOW: $P / 0$	DD?
(5)	$\Box P / 0$	2,wt(0)
(6)	$\forall i\{0 \prec i \rightarrow [P / i]\}$	5,wt(□)
(7)	???	

Example 2

(1)	SHOW: $\Box P \rightarrow \Box\Box P$	CD
(2)	$\Box P$	As
(3)	SHOW: $\Box\Box P$	wt(0)
(4)	SHOW: $\Box\Box P / 0$	kwt(□)
(5)	SHOW: $\forall i\{0 \prec i \rightarrow [\Box P / i]\}$	UCD
(6)	$0 \prec 1$	As
(7)	SHOW: $\Box P / 1$	kwt(□)
(8)	SHOW: $\forall i\{1 \prec i \rightarrow [P / i]\}$	UCD
(9)	$1 \prec 2$	As
(10)	SHOW: $P / 2$	DD?
(11)	$\Box P / 0$	2,wt(0)
(12)	$\forall i\{0 \prec i \rightarrow [P / i]\}$	11,kwt(□)
(13)	???	

6. System K and Kripkean World Theory

Just as there is a simple relation between Modal System L and Leibnizian World Theory, there is a corresponding relation between Modal System K and Kripkean World Theory.

(T) Concentrating on just those formulas shared by System K and WT(K), a formula is a thesis of System K iff it is a thesis of WT(K).

C. Counter-Models in System K

1. Valuations in System K

Having presented a method of demonstrating validity in System K, we next present a companion technique for demonstrating invalidity in System K Just as we did with System L, we begin with the semantics of Indexed SL, and append to it clauses pertaining to the modal operators. The complication is that, in relative modal logic, the semantics makes reference to an accessibility relation $\prec$.

(d2) Let $\mathcal{L}$ be a language, and let S be the associated set of formulas of $\mathcal{L}$. Then a ***valuation*** on $\mathcal{L}$ is, by definition, any function from S into {T,F}. In other words, a valuation is a function that assigns a truth-value to each formula of $\mathcal{L}$.

(d3) Let $\mathcal{L}$ be the language of modal sentential logic; let S be the associated set of formulas of $\mathcal{L}$; let Ω be a set of indices. Then an ***indexed-valuation*** on $\mathcal{L}$ is, by definition, any function from $S\times\Omega$ into {T,F}. In other words, an indexed valuation is a function that assigns a truth-value to each ordered pair ϕ/i where ϕ is a formula and i is an index.[7]

(d4) A ***K-model*** is, by definition, a structure $\langle\ \Omega, \prec, \omega_0, \langle p_1,p_2,\ldots\rangle\ \rangle$, where Ω (omega) is a non-empty set, $\prec$ is a relation on Ω, ω_0 is a (privileged) element of Ω, and $\langle p_1,p_2,\ldots\rangle$ is an infinite sequence of subsets of Ω.

In a K-model (originally proposed by Kripke, 1958), Ω is the set of indices (possible worlds), $\prec$ is the accessibility relation, and ω_0 is the "original" or "actual" world. On the other hand, the sequence $\langle p_1,p_2,\ldots\rangle$ serve to interpret the sequence $\langle A_1,A_2,\ldots\rangle$ of atomic formulas of $\mathcal{L}$.

(d5) Let $\mathcal{L}$ be the language of modal sentential logic; let S be the associated set of formulas of $\mathcal{L}$. Let $\langle\ \Omega, \prec, \omega_0, \langle p_1,p_2,\ldots\rangle\ \rangle$ be a K-model. Then there is an ***associated indexed valuation*** w on $\mathcal{L}$ satisfying the following conditions for every element i of Ω.

(0)	$w(A_m / i) = \mathrm{T}$	iff	$i \in p_m$
(1)	$w(\sim\phi / i) = \mathrm{T}$	iff	$w(\phi / i) = \mathrm{F}$
(2)	$w(\phi\&\psi / i) = \mathrm{T}$	iff	$w(\phi / i) = \mathrm{T}$ and $\upsilon(\psi / i) = \mathrm{T}$
(3)	$w(\phi\vee\psi / i) = \mathrm{T}$	iff	$w(\phi / i) = \mathrm{T}$ and/or $\upsilon(\psi / i) = \mathrm{T}$
(4)	$w(\phi\rightarrow\psi / i) = \mathrm{T}$	iff	$w(\phi / i) \leqslant \upsilon(\psi / i)$
(5)	$w(\phi\leftrightarrow\psi / i) = \mathrm{T}$	iff	$w(\phi / i) = \upsilon(\psi / i)$
(6)	$w(\Box\phi / i) = \mathrm{T}$	iff	$w(\phi / j) = \mathrm{T}$, for every j such that $i\prec j$
(7)	$w(\Diamond\phi / i) = \mathrm{T}$	iff	$w(\phi / j) = \mathrm{T}$, for at least one j such that $i\prec j$

(d6) Let $\mathcal{L}$ and S be as before. Let υ be a valuation on $\mathcal{L}$. Then υ is a **K-admissible valuation** if and only if there is a K-model with an associated indexed valuation w such that, for every formula ϕ, $\upsilon(\phi) = w(\phi/\omega_0)$.

(d7) Let $\mathcal{L}$ and S be as before. Let $\langle P_1,\ldots,P_m/C\rangle$ be an argument in $\mathcal{L}$. Then a ***K-counter-model*** to $\langle P_1,\ldots,P_m/\mathrm{C}\rangle$ is any K-admissible valuation υ such that:
$\upsilon(P_1) = \mathrm{T}, \ldots, \upsilon(P_m) = \mathrm{T}$, and $\upsilon(C)=\mathrm{F}$.

[7] We adopt an alternative decorative notation for ordered pairs so that α/β is the ordered pair whose first component is α and whose second component is β. We do this so we can read 'α/β' as "alpha at beta".

2. Counter-Models in System K

We now consider a paper-and-pencil way to construct counter-models in System K. As an example, consider the following argument.

$$\Box(P \vee Q) \;/\; (\Box P \vee \Box Q)$$

We have already shown that this argument is invalid in System L. It is therefore invalid in the weaker System K. Nevertheless, let us examine how the counter-model construction goes, just for practice.

First, we start our construction at index 0, and we assign T to the premise and F to the conclusion, which gives us the following situation.

	□	(	P	∨	Q	)	/	(	□	P	∨	□	Q	)
0	T										F			

Next, we apply the truth conditions to index 0 – where we can. First, since $(\Box P \vee \Box Q)$ is F/0, $\Box P$ is F/0, and $\Box Q$ is F/0. On the other hand, although $\Box(P \vee Q)$ is T at 0, it does not follow that $(P \vee Q)$ is T at 0! This is already a big difference between System K and System L. This yields the following.

	□	(	P	∨	Q	)	/	(	□	P	∨	□	Q	)
0	T								F		F	F		

Next, since $\Box P$ is F/0, P must be F at *some* index. To achieve this, we "create" a new index, labeled '01', at which P is F. Similarly, $\Box Q$ is F/0, so Q must be F at some index. For this, we create a new index, labeled '02', at which Q is F. The tacit understanding is that 0 sees the two offspring indices 01 and 02. This yields the following. [Notice: if a formula occurs twice on a line, its truth value is written twice.]

	□	(	P	∨	Q	)	/	(	□	P	∨	□	Q	)
0	T								F		F	F		
01			F							F				
02					F								F	

But since $\Box(P \vee Q)$ is T/0, $(P \vee Q)$ must be T/01, and T/02, since 0 sees both 01 and 02. Applying this to the above diagram, we obtain:

$\Box$ (P $\vee$ Q) / ($\Box$ P $\vee$ $\Box$ Q)

	$\Box$	P	$\vee$	Q	$\Box$	P	$\vee$	$\Box$	Q
0	T				F		F	F	
01		F	T			F			
02			T	F					F

Next, applying the truth-function for $\vee$ yields the following.

$\Box$ (P $\vee$ Q) / ($\Box$ P $\vee$ $\Box$ Q)

	$\Box$	P	$\vee$	Q	$\Box$	P	$\vee$	$\Box$	Q
0	T				F		F	F	
01		F	T	T		F			T
02		T	T	F		T			F

Finally, we note that the table can be completed in a manner that is consistent with the semantic rules. One way to do this (but not the only way) is as follows.

$\Box$ (P $\vee$ Q) / ($\Box$ P $\vee$ $\Box$ Q)

	$\Box$	P	$\vee$	Q	$\Box$	P	$\vee$	$\Box$	Q
0	T	F	F	F	F	F	F	F	F
01	T	F	T	T	F	F	F	F	T
02	T	T	T	F	F	T	F	F	F

When we are done, we can summarize the construction as follows. First, the K-model that provides the counter-model is $\langle \Omega, \prec, \omega_0, \langle p_1,p_2 \rangle \rangle$, where $\Omega = \{0, 01, 02\}$, $\prec = \{0\rightarrow 01, 0\rightarrow 02\}$,[8] $\omega_0 = 0$, and $\langle p_1,p_2 \rangle = \langle \{02\},\{01\} \rangle$. We also understand P to be the first atomic formula, and Q to be the second atomic formula.[9]

Now, let us consider an argument that is valid according to System L, but is not valid according to System K.

$\Diamond\Box$P / $\Box\Diamond$P

[8] Once again, we adopt a decorative variation of ordered pair notation. In this case $\alpha\rightarrow\beta$ is the ordered pair whose first component is α and whose second component is β. We read '$\alpha\rightarrow\beta$' as "α bears the relation to β", which in this particular case means that α sees β.

[9] The remaining atomic formulas are ignored here.

As before, we begin with the following truth table for index 0.

	◇	□	P	/	□	◇	P
0	T				F		

Our next step is to apply ◇O and ~□O, which produces the following situation.

	◇	□	P	/	□	◇	P
0	T				F		
01		T					
02						F	

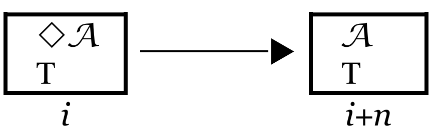
Curiously, we are done; there are no more rules to apply. We can fill in the truth value of P at 0, 01, and 02 any way we wish. The key to this freedom is that the only accessibility among the worlds is that explicitly associated with the offspring relation: 0 sees 01; 0 sees 02; but that is it!

This may seem odd, but it is in keeping with just how weak System K is. Later, we examine systems that are stronger than K but weaker than L.

3. Pictorial Presentation of Rules for Constructing Counter-Models in K

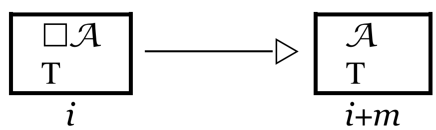
◇O

If ◇$\mathcal{A}$ is T at index i, then $\mathcal{A}$ is T at some (new!) index $i+n$.



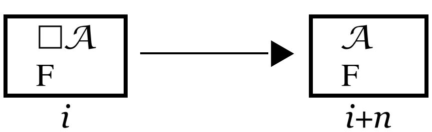
□O

If □$\mathcal{A}$ is T at index i, then $\mathcal{A}$ is T at every index accessible to i.



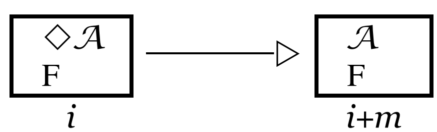
~□O

If □$\mathcal{A}$ is F at index i, then $\mathcal{A}$ is F at some (new!) index $i+n$.



~◇O

If ◇$\mathcal{A}$ is F at index i, then $\mathcal{A}$ is F at every index accessible to i.



——▶ indicates "creating" a "new" *accessible* world.

——▷ indicates discharging a rule at every existing ("old") *accessible* world.

4. The Relation Between Derivations and Counter-Models in System K

Once again, the astute reader has probably noticed that the rules for constructing counter-models parallel the rules for constructing derivations. As before, this is no accident.

Consider our earlier example.

□(P ∨ Q) / □P ∨ □Q

This time, let us construct a *partial* derivation in System K of the conclusion from the premises.

(1)	□(P ∨ Q)	/0	Pr
(2)	~: □P ∨ □Q	/0	ID
(3)	~(□P ∨ □Q)	/0	As
(4)	~: ※	/✳	DD
(5)	~□P	/0	3,SL
(6)	~□Q	/0	3,SL
(7)	P ∨ Q	/0	1,□O
(8)	~P	/01	5,~□O
(9)	~Q	/02	6,~□O
(10)	P ∨ Q	/01	1,□O
(11)	P ∨ Q	/02	1,□O
(12)	P	/01	8,10,SL
(13)	Q	/02	9,11,SL

Once again, we can read the counter-model from the partial derivation.

Next, consider our other example from above

◇□P / □◇P

and consider the following partial derivation.

(1)	◇□P	/0	Pr
(2)	~: □◇P	/0	ID
(3)	~□◇P	/0	As
(4)	~: ※	/✳	DD
(5)	□Q	/01	1,◇O
(6)	~◇P	/02	3,~□O

As above, we can read the counter-model from this partial derivation.

D. Exercises

1. Exercises In System K

Directions: for each of the following, construct a formal derivation of the conclusion (marked by '**/**') from the premises (if any) in Modal System K. In problems in which two formulas are separated by '**//**', construct a derivation of each formula from the other.

1. **/** □[P → (P ∨ Q)]
2. **/** □P → □(P ∨ Q)
3. **/** □(~□P ↔ ◇~P)
4. **/** □(~◇P ↔ □~P)
5. **/** □(P ∨ ~P)
6. □(P → Q) **//** ~◇(P & ~Q)
7. □(P → Q) **;** □P **/** □Q
8. **/** □(P → Q) → (□P → □Q)
9. □(P ↔ Q) **/** □P ↔ □Q
10. □(P & Q) **//** □P & □Q
11. ~◇(P ∨ Q) **/** ~◇P & ~◇Q
12. ~◇(P & Q) **//** □(P → ~Q)
13. ◇(P → Q) **//** □P → ◇Q
14. □P ∨ □Q **/** □(P ∨ Q)
15. □(P → Q) **;** ◇P **/** ◇Q
16. **/** □(P → Q) → (◇P → ◇Q)
17. ◇(P ∨ Q) **//** ◇P ∨ ◇Q
18. ◇(P & Q) **/** ◇P & ◇Q
19. ◇P & □Q **/** ◇(P & Q)
20. □(P ∨ Q) **/** □P ∨ ◇Q
21. □(P → ~P) **//** ~◇P
22. □(~P → P) **//** □P
23. ~◇P **/** □(P → Q)
24. □Q **/** □(P → Q)
25. □P **;** ◇~Q **/** ~□(P → Q)
26. □(P ↔ Q) **/** □P ↔ □Q
27. □(P ↔ Q) **/** ◇P ↔ ◇Q
28. □(P ∨ Q) **;** ◇~P **/** ◇Q
29. □(P ∨ Q) **;** ~◇P **/** □Q
30. □P **;** □Q **/** □(P ↔ Q)
31. ~◇P **;** ~◇Q **/** □(P ↔ Q)
32. □(P → Q) **;** ◇~Q **/** ~□P
33. ~□(P → Q) **/** ◇P & ◇~Q
34. ~□(P ↔ Q) **/** ◇P ∨ ◇Q
35. □(P → Q) **;** □(P → ~Q) **/** ~◇P
36. □(P → Q) **;** □(~P → Q) **/** □Q
37. ◇(P → Q) **;** □P **/** ◇Q
38. **/** ◇(P → Q) → (□P → ◇Q)
39. □◇P **;** ◇□(P → Q) **/** ◇◇Q
40. ◇P **/** □Q → ◇Q
41. □(P → Q) **//** □(~Q → ~P)
42. □(P → Q) **//** □[P → (P → Q)]
43. □(P → Q) **//** □(P → (P & Q))
44. □(P → Q) **//** □[(P ∨ Q) → Q]
45. □[P → (Q&R)] **//** □(P→Q) & □(P→R)
46. □[(P∨Q) → R] **//** □(P→R) & □(Q→R)
47. □(P→Q) ∨ □(P→R) **/** □[P → (Q∨R)]
48. □(P → Q) **;** □(Q → R) **/** □(P → R)
49. □[P → (Q → R)] **/** □(P→Q) → □(P→R)
50. □(P → Q) **;** ◇(P & R) **/** ◇(Q & R)
51. ◇□P **;** □◇Q **/** ◇◇(P & Q)
52. □(P → Q) **;** ◇(P & R) **/** ◇(Q & R)
53. □P → □Q **/** ◇R → ◇(P → Q)

2. Exercises in WT(K)

Directions: For each argument form in Part 1, construct a derivation in Kripkean World Theory.

3. Exercises in Counter-Models

For each of the following argument forms, which are all valid in System L, construct a counter-model in System K.

1. $/ \Diamond P \rightarrow \Box\Diamond P$
2. $/ \Diamond\Box P \rightarrow \Box P$
3. $/ \Box P \rightarrow \Box\Box P$
4. $/ \Diamond\Diamond P \rightarrow \Diamond P$
5. $\Box(P \leftrightarrow Q) / \Box(\Box P \leftrightarrow \Box Q)$
6. $\Box(P \leftrightarrow Q) / \Box(\Diamond P \leftrightarrow \Diamond Q)$
7. $\Box(P \rightarrow \Diamond Q) / \Box(\Diamond P \rightarrow \Diamond Q)$
8. $\Diamond(P \rightarrow \Box Q)$ **;** $\Box P / \Diamond Q$
9. $\Diamond\Box P / \Diamond(P \vee Q)$
10. $\Box P \vee \Box Q / \Box(\Box P \vee \Box Q)$
11. $\Box(\Box P \rightarrow Q) / \Box(\Box P \rightarrow \Box Q)$
12. $\Diamond P$ **;** $\Box Q / \Diamond(P \,\&\, \Box Q)$
13. $\Box(P \vee \Box Q) // \Box P \vee \Box Q$
14. $\Box(P \vee \Diamond Q) // \Box P \vee \Diamond Q$
15. $\Diamond(P \,\&\, \Box Q) // \Diamond P \,\&\, \Box Q$
16. $\Diamond(P \,\&\, \Diamond Q) // \Diamond P \,\&\, \Diamond Q$
17. $\Box(P \rightarrow \Box Q) // \Diamond P \rightarrow \Box Q$
18. $\Box(P \rightarrow \Diamond Q) // \Diamond P \rightarrow \Diamond Q$
19. $\Diamond(P \rightarrow \Box Q) // \Box P \rightarrow \Box Q$
20. $\Diamond(P \rightarrow \Diamond Q) // \Box P \rightarrow \Diamond Q$
21. $\Box(P \leftrightarrow \Box Q) / \Box P \leftrightarrow \Box Q$
22. $\Box(P \leftrightarrow \Diamond Q) / \Box P \leftrightarrow \Diamond Q$
23. $\Diamond P \leftrightarrow \Box Q / \Diamond(P \leftrightarrow \Box Q)$
24. $\Diamond P \leftrightarrow \Diamond Q / \Diamond(P \leftrightarrow \Diamond Q)$
25. $/ \Box(\Box P \rightarrow \Box Q) \vee \Box(\Box Q \rightarrow \Box P)$

4. Answers to Selected Exercises

1. System K

Note that the annotations ‘new’ and ‘old’ are *optional*; they are included here in order to emphasize the difference between $\Box$O/$\Diamond$I (old) and $\Box$D/$\Diamond$O (new).

#8

(1)	~~SHOW~~: $\Box(P \rightarrow Q) \rightarrow (\Box P \rightarrow \Box Q)$	/0	CD	
(2)	│ $\Box(P \rightarrow Q)$	/0	As	
(3)	│ ~~SHOW~~: $\Box P \rightarrow \Box Q$	/0	CD	
(4)	│ │ $\Box P$	/0	As	
(5)	│ │ ~~SHOW~~: $\Box Q$	/0	ND	
(6)	│ │ ~~SHOW~~: Q	/01	DD	new
(7)	│ │ │ $P \rightarrow Q$	/01	2,$\Box$O	old
(8)	│ │ │ P	/01	4,$\Box$O	old
(9)	│ │ │ Q	/01	7,8,SL	

#13a

(1)	$\Diamond(P \rightarrow Q)$	/0	Pr	
(2)	~~SHOW~~: $\Box P \rightarrow \Diamond Q$	/0	CD	
(3)	│ $\Box P$	/0	As	
(4)	│ ~~SHOW~~: $\Diamond Q$	/0	DD	
(5)	│ │ $P \rightarrow Q$	/01	1,$\Diamond$O	new
(6)	│ │ P	/01	3,$\Box$O	old
(7)	│ │ Q	/01	5,6,SL	
(8)	│ │ $\Diamond Q$	/0	7,$\Diamond$I	01 old

#13b

(1)	□P → ◇Q	/0	Pr	
(2)	~~SHOW~~: ◇(P → Q)	/0	ID	
(3)	│ ~◇(P → Q)	/0	As	
(4)	│ ~~SHOW~~: ※	/*	10,12,SL	
(5)	│ │ ~~SHOW~~: □P	/0	ND	
(6)	│ │ ~~SHOW~~: P	/01	DD	new
(7)	│ │ │ ~(P → Q)	/01	3,~◇O	old
(8)	│ │ │ P	/01	7,SL	
(9)	│ │ ◇Q	/0	1,5,SL	
(10)	│ │ Q	/01	9,◇O	new!
(11)	│ │ ~(P → Q)	/01	3,~◇O	old
(12)	│ │ ~Q	/01	11,SL	

#19

(1)	◇P & □Q	/0	Pr	
(2)	~~SHOW~~: ◇(P & Q)	/0	DD	
(3)	│ P	/01	1a,◇O	new
(4)	│ Q	/01	1b,□O	old
(5)	│ P & Q	/01	3,4,SL	
(6)	│ ◇(P & Q)	/0	5,◇I	01 old

#23

(1)	~◇P	/0	Pr	
(2)	~~SHOW~~: □(P → Q)	/0	ND	
(3)	~~SHOW~~: P → Q	/01	CD	new
(4)	│ P	/01	As	
(5)	│ ~~SHOW~~: Q	/01	4,6,SL	
(6)	│ ~P	/01	1,~◇O	old

#33

(1)	~□(P → Q)	/0	Pr	
(2)	~~SHOW~~: ◇P & ◇~Q	/0	5,6,SL	
(3)	│ ~(P → Q)	/01	1,~□O	new
(4)	│ P & ~Q	/01	3,SL	
(5)	│ ◇P	/0	4a,◇I	01 old
(6)	│ ◇~Q	/0	4b,◇I	01 old

#34

(1)	~□(P ↔ Q)	/0	Pr	
(2)	~~SHOW~~: ◇P ∨ ◇Q	/0	∨D	
(3)	│ ~◇P	/0	As	
(4)	│ ~◇Q	/0	As	
(5)	│ ~~SHOW~~: ※	/*	6,7,8,SL	
(6)	│ │ ~(P ↔ Q)	/01	1,~□O	new
(7)	│ │ ~P	/01	3,~◇O	old
(8)	│ │ ~Q	/01	4,~◇O	old

#35

(1)	□(P → Q)	/0	Pr	
(2)	□(P → ~Q)	/0	Pr	
(3)	~~SHOW~~: ~◇P	/0	ID	
(4)	◇P	/0	As	
(5)	~~SHOW~~: ※	/*	DD	
(6)	P	/01	4,◇O	new
(7)	P → Q	/01	1,□O	old
(8)	Q	/01	7,8,SL	
(9)	P → ~Q	/01	2,□O	old
(10)	~Q	/01	6,9,SL	
(11)	※	/*	8,10,SL	

#37

(1)	◇(P → Q)	/0	Pr	
(2)	□P	/0	Pr	
(3)	~~SHOW~~: ◇Q	/0	6,◇I	01 old
(4)	P → Q	/01	1,◇O	new
(5)	P	/01	2,□O	old
(6)	Q	/01	4,5,SL	

#39

(1)	□◇P	/0	Pr	
(2)	◇□(P → Q)	/0	Pr	
(3)	~~SHOW~~: ◇◇Q	/0	DD	
(4)	□(P → Q)	/01	2,◇O	new
(5)	◇P	/01	1,□O	old
(6)	P	/012	5,◇O	new
(7)	P → Q	/012	4,□O	old
(8)	Q	/012	6,7,SL	
(9)	◇Q	/01	8,◇I	012old
(10)	◇◇Q	/0	9,◇I	01 old

#40

(1)	◇P	/0	Pr	
(2)	~~SHOW~~: □Q → ◇Q	/0	CD	
(3)	□Q	/0	As	
(4)	~~SHOW~~: ◇Q	/0	DD	
(5)	P	/01	1,◇O	new
(6)	Q	/01	3,□O	old
(7)	◇Q	/0	6,◇I	01 old

#42a

(1)	□(P → Q)	/0	Pr	
(2)	~~SHOW~~: □(P → (P → Q))	/0	ND	new
(3)	~~SHOW~~: P → (P → Q)	/01	CD	
(4)	P	/01	As	
(5)	~~SHOW~~: P → Q	/01	CD	
(6)	P	/01	As	
(7)	~~SHOW~~: Q	/01	DD	
(8)	P → Q	/01	1,□O	old
(9)	Q	/01	6,8,SL	

#49

(1)	□(P → (Q → R))	/0	Pr	
(2)	~~SHOW~~: □(P→Q) → □(P→R)	/0	CD	
(3)	□(P → Q)	/0	As	
(4)	~~SHOW~~: □(P → R)	/0	ND	
(5)	~~SHOW~~: P → R	/01	CD	
(6)	P	/01	As	new
(7)	~~SHOW~~: R	/01	DD	
(8)	P → (Q → R)	/01	1,□O	old
(9)	P → Q	/01	3,□O	old
(10)	R	/01	6,8,9,SL	

#50

(1)	□(P → Q)	/0	Pr	
(2)	◇(P & R)	/0	Pr	
(3)	~~SHOW~~: ◇(Q & R)	/0	DD	
(4)	P & R	/01	2,◇O	new
(5)	P → Q	/01	1,□O	old
(6)	Q & R	/01	4,5,6,SL	
(7)	◇(Q & R)	/0	6,◇I	01 old

#51

(1)	◇□P	/0	Pr	
(2)	□◇Q	/0	Pr	
(3)	~~SHOW~~: ◇◇(P & Q)	/0	DD	
(4)	□P	/01	1,◇O	new
(5)	◇Q	/01	2,□O	old
(6)	Q	/012	5,◇O	new
(7)	P	/012	4,□O	old
(8)	P & Q	/012	6,7,SL	
(9)	◇(P & Q)	/01	8,◇I	012old
(10)	◇◇(P & Q)	/0	9,◇I	01 old

#52

(1)	□(P → Q)	/0	Pr	
(2)	◇(P & R)	/0	Pr	
(3)	~~SHOW~~: ◇(Q & R)	/0	DD	
(4)	P & R	/01	2,◇O	new
(5)	P → Q	/01	1,□O	old
(6)	Q & R	/01	4,5,SL	
(7)	◇(Q & R)	/0	6,◇I	01 old

#53

(1)	□P → □Q	/0	Pr	
(2)	~~SHOW~~: ◇R → ◇(P → Q)	/0	CD	
(3)	│ ◇R	/0	As	
(4)	│ ~~SHOW~~: ◇(P → Q)	/0	ID	
(5)	│ │ ~◇(P → Q)	/0	As	
(6)	│ │ ~~SHOW~~: ※	/*	DD	
(7)	│ │ │ ~~SHOW~~: □P	/0	ND	
(8)	│ │ │ │ ~~SHOW~~: P	/01	DD	new
(9)	│ │ │ │ │ ~(P → Q)	/01	5,~◇O	old
(10)	│ │ │ │ │ P	/01	9,SL	
(11)	│ │ │ □Q	/0	1,7,SL	
(12)	│ │ │ R	/02	3,◇O	new
(13)	│ │ │ Q	/02	11,□O	old
(14)	│ │ │ ~(P → Q)	/02	5,~◇O	old
(15)	│ │ │ ~Q	/02	14,SL	
(16)	│ │ │ ※	/*	13,15,SL	

2. System WT(K)

#13a

(1)	◇(P → Q)	Pr
(2)	◇(P → Q) / 0	1, wt(0)
(3)	~~SHOW~~: □P → ◇Q	wt(0)
(4)	~~SHOW~~: (□P → ◇Q) / 0	wt(→)
(5)	~~SHOW~~: [□P / 0] ⟶ [◇Q) / 0]	CD
(6)	│ □P / 0	As
(7)	│ ~~SHOW~~: ◇Q / 0	DD
(8)	│ │ ∃*i*{0≺*i* & [P → Q / *i*]	2,kwt(◇)
(9)	│ │ 0≺1	8,∃&O
(10)	│ │ P→Q / 1	8,∃&O
(11)	│ │ [P / 1] ⟶ [Q / 1]	10,wt(→)
(12)	│ │ ∀*i*{0≺*i* ⟶ [P / *i*]}	6,kwt(□)
(13)	│ │ P / 1	9,12,QL
(14)	│ │ Q / 1	11,13,SL
(15)	│ │ ∃*i*{0≺*i* & [Q / *i*]}	9,14,QL
(16)	│ │ ◇Q / 0	15,kwt(◇)

#13b

(1)	□P → ◇Q	Pr
(2)	(□P → ◇Q) / 0	1,wt(0)
(3)	[□P / 0] → [◇Q / 0]	2,wt(→)
(4)	~~SHOW~~: ◇(P → Q)	wt(0)
(5)	~~SHOW~~: ◇(P → Q) / 0	kwt(◇)
(6)	~~SHOW~~: ∃*i*{0≺*i* & [(P→Q) / *i*]}	ID
(7)	~∃*i*{0≺*i* & [(P→Q) / *i*]}	As
(8)	~~SHOW~~: ※	DD
(9)	~~SHOW~~: □P / 0	kwt(□)
(10)	~~SHOW~~: ∀*i*{0≺*i* → [P / *i*]}	UCD
(11)	0≺1	As
(12)	~~SHOW~~: P / 1	ID
(13)	~[P / 1]	As
(14)	~~SHOW~~: ※	14,16,SL
(15)	~~SHOW~~: (P → Q) / 1	wt(→)
(16)	~~SHOW~~: [P / 1] → [Q / 1]	12,SL
(17)	~[(P → Q) / 1]	6,10,QL
(18)	◇Q / 0	3,9,SL
(19)	∃*i*{0≺*i* & [Q / *i*]}	18,wt(◇)
(20)	0≺1	19,∃&O
(21)	Q / 1	19,∃&O
(22)	[P / 1] → [Q / 1]	21,SL
(23)	(P → Q) / 1	22,wt(→)
(24)	※	7,20,23,QL

#19

(1)	◇P & □Q	Pr
(2)	(◇P & □Q) / 0	1,wt(0)
(3)	[◇P / 0] & [□Q / 0]	2,wt(&)
(4)	~~SHOW~~: ◇(P & Q)	wt(0)
(5)	~~SHOW~~: ◇(P & Q) / 0	kwt(◇)
(6)	~~SHOW~~: ∃*i*{0≺*i* & [P & Q) / *i*]}	DD
(7)	∃*i*{0≺*i* & [P / *i*]}	3a,wt(◇)
(8)	0≺1	7,∃&O
(9)	P / 1	7,∃&O
(10)	∀*i*{0≺*i* → [Q / *i*]}	3b,wt(□)
(11)	Q / 1	8,10,QL
(12)	[P / 1] & [Q / 1]	9,11,SL
(13)	(P & Q) / 1	12,wt(&)
(14)	∃*i*{0≺*i* & [(P & Q) / *i*]}	8,13,QL

#51

(1)	◇□P	Pr
(2)	□◇Q	Pr
(3)	◇□P / 0	1,wt(0)
(4)	□◇Q / 0	2,wt(0)
(5)	~~SHOW~~: ◇◇(P & Q)	wt(0)
(6)	~~SHOW~~: ◇◇(P & Q) / 0	kwt(◇)
(7)	~~SHOW~~: ∃i{0≺i & [◇(P & Q) / i]}	DD
(8)	∃i{0Ri & [□P / i]}	3,kwt(◇)
(9)	0≺1	8,∃&O
(10)	□P / 1	8,∃&O
(11)	∀i{0≺i → [◇Q / i]}	4,kwt(□)
(12)	◇Q / 1	9,11,QL
(13)	∃i{1≺i & [Q / i]}	12,kwt(◇)
(14)	1≺2	13,∃&O
(15)	Q / 2	13,∃&O
(16)	∀i{1≺i → [P / i]}	10,kwt(□)
(17)	P / 2	14,16,QL
(18)	[P / 2] & [Q / 2]	15,17,SL
(19)	(P & Q) / 2	18,wt(&)
(20)	∃i{1≺i & [(P & Q) / i]}	14,19,QL
(21)	◇(P & Q) / 1	20,kwt(◇)
(22)	∃i{0≺i & [◇(P & Q) / i]}	9,21,QL

3. Counter-Models in K

#1 ◇ P → □ ◇ P

	◇	P	→	□	◇	P
0	T		F	F		
01		T				

#7 □ (P → ◇ Q) / □ (◇ P → ◇ Q)

	□	P	→	◇	Q
0	T				
01					

	□	◇	P	→	◇	Q
0	F					
01		T		F	F	

#3 □ P → □ □ P

	□	P	→	□	□	P
0	T		F	F		
01		T			F	T

#12 ◇ P ; □ Q / ◇ (P & □ Q)

	◇	P
0	T	
01		T

	□	Q
0	T	
01		T

	◇	P	&	□	Q
0	F				
01		T	F	F	