For as anyone may feel certain that a chain will hold when assured that each separate link is of good material and that it clasps well the two neighboring links, . . . so we may be certain of the accuracy of reasoning when the subject matter is good; that is to say when nothing doubtful enters into it, and when the form consists in a perpetual concatenation of truths.
INTRODUCTION TO DEDUCTIVE LOGIC PART II: Argument Evaluation

TABLE OF CONTENTS

10 Logical Operator Rules
5 Quantifier Rules

I. FOCUS ON STRUCTURE

| Deductive Argument Evaluation: Validity and Invalidity | 5 |
| Valid vs Invalid: Definitions | 6 |
| Demonstrating Validity and Invalidity: Truth Tables | 7 |
| Setting up truth tables for arguments: A Full Truth Table Method | 9 |
| Reading Truth Tables to Determine Validity or Invalidity | 10 |
| Truth Table Exercises |
| Tautology Test: A Second Truth Table Method | 14 |
| A Super Short Cut method | 16 |

II. FOCUS ON CONTENT

Consistency/Inconsistency

III. FORMAL DEDUCTIVE PROOF: THE GOLD STANDARD

QUANTIFIER RULES

| The issue of bound and free variables | 26 |
| Quantifier Rules with Restrictions | 28 |
| Changing Quantifiers: A Replacement Rule | 30 |

LOGICAL OPERATOR RULES

| Conjunction Rules | 34 |
| Deductive Proof: The Two-Column Method | 37 |
| Beginning a Proof | 37 |
| Conditional Rules | 40 |
| Using Assumption Rules | 45 |
| Bi-conditional Rules | 48 |
| Working with Disjunctions | 53 |
| Working with Negations | 59 |
| Application Challenge: Proving Double Negation | 60 |

DEVELOPING PROOFS: SKILLS AND STRATEGIES

| Bonus Bank Assignment: Proving Theorems | 68 |
| Developing and Expanding Proof Skills: Practice Exercises | 69-77 |

IV. ANSWERS TO SELECTED EXERCISES

© Sharon Rowe, PhD., 2013-14
### 10 Logical Operator Rules

<table>
<thead>
<tr>
<th>Rule Type</th>
<th>Symbol</th>
<th>Logical Operator Formula</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conjunction Elimination</td>
<td>( \land )</td>
<td>( p \land q \rightarrow p )</td>
<td>Elimination of Conjunction</td>
</tr>
<tr>
<td>Conjunction Introduction</td>
<td>( \land )</td>
<td>( p \rightarrow q \rightarrow p \land q )</td>
<td>Introduction of Conjunction</td>
</tr>
<tr>
<td>Conditional Eliminating</td>
<td>( \rightarrow )</td>
<td>( p \rightarrow q \rightarrow p )</td>
<td>Elimination of Conditional</td>
</tr>
<tr>
<td>Conditional Introduction</td>
<td>( \rightarrow )</td>
<td>( p \rightarrow q \rightarrow p \rightarrow q )</td>
<td>Introduction of Conditional</td>
</tr>
<tr>
<td>Disjunction Introduction</td>
<td>( \lor )</td>
<td>( p \lor q \rightarrow p \lor q )</td>
<td>Introduction of Disjunction</td>
</tr>
<tr>
<td>Disjunction Elimination</td>
<td>( \lor )</td>
<td>( p \lor q \rightarrow p \lor q \rightarrow r )</td>
<td>Elimination of Disjunction</td>
</tr>
<tr>
<td>Bi-Conditional Elimination</td>
<td>( \iff )</td>
<td>( p \iff q \rightarrow p \iff q )</td>
<td>Elimination of Bi-Conditional</td>
</tr>
<tr>
<td>Bi-Conditional Introduction</td>
<td>( \iff )</td>
<td>( p \iff q \rightarrow p \iff q \rightarrow p \iff q )</td>
<td>Introduction of Bi-Conditional</td>
</tr>
<tr>
<td>Negation Elimination/Introduction</td>
<td>( \sim )</td>
<td>( \sim p \rightarrow p )</td>
<td>Elimination/Introduction of Negation</td>
</tr>
</tbody>
</table>
Quantifier Rules:

<table>
<thead>
<tr>
<th>Universal Elimination / ( \forall E )</th>
<th>Existential Introduction / ( \exists I )</th>
</tr>
</thead>
<tbody>
<tr>
<td>((\forall x) Ax )</td>
<td>( Aa ) (where ( a ) is any individual constant)</td>
</tr>
<tr>
<td>( \therefore Aa ) (where ( a ) is any individual constant)</td>
<td>( \therefore (\exists x) Ax )</td>
</tr>
</tbody>
</table>

\( \forall E \) has no restrictions

| Universal Introduction / \( \forall I \) |
|--------------------------------..........|
| \( Ar \) (Where \( r \) is a randomly selected individual) | \( (\exists x) Ax \) |
| \( \therefore (\forall x) Ax \) |

Restrictions:

1) This rule may **not** be used to generalize from a specifically named/identified individual (temporary or specified).
2) The universal quantifier may be validly inferred **only** from a randomly selected individual. In this course, the symbol “\( r \)” will indicate a randomly selected individual **unless** otherwise indicated by a language key.

<table>
<thead>
<tr>
<th>Logical Equivalence Rule/ ( LE )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \sim (\forall x) Ax \equiv (\exists x) \sim Ax )</td>
</tr>
<tr>
<td>( \sim (\exists x) Ax \equiv (\forall x) \sim Ax )</td>
</tr>
</tbody>
</table>

This rule is a replacement rule. A replacement rule lets you replace any occurrence of an expression of like form on one side of the equivalence sign with the expression on the other side.

<table>
<thead>
<tr>
<th>Existential Elimination Rule / ( \exists E )</th>
</tr>
</thead>
<tbody>
<tr>
<td>((\exists x) Ax )</td>
</tr>
<tr>
<td>( \therefore At )</td>
</tr>
</tbody>
</table>

Restrictions:

1) You may not use “\( t \)” to introduce an individual constant already named in the proof or appearing in the conclusion. (In other words, “\( t \)” will be used only once in any proof.)
2) The temporary name must be removed before the proof is validly complete. This will occur when you reintroduce the existential quantifier using the \( \exists I \) rule.

<table>
<thead>
<tr>
<th>LE Rule/ Special Case: Categorical Statements</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \sim(\forall x)(Ax \supset Bx) \equiv (\exists x)(Ax \cdot \sim Bx) )</td>
</tr>
<tr>
<td>( \sim(\exists x)(Ax \cdot Bx) \equiv (\forall x)(Ax \supset \sim Bx) )</td>
</tr>
</tbody>
</table>

In this course, this rule will apply to standard form categorical statements. Note the basic pattern for categorical statements is negated. Notice where the negation (\( \sim \)) sign moves and the change of the logical operator to coordinate with the quantifier.
FOCUS ON ARGUMENT STRUCTURE

Reasoning begins with statements we believe are true and draws forward from them, or justifies other statements. We claim that the statements we draw forward are true solely because the premise statements from which they are drawn, are stated to be true. In this way reasoning expands our knowledge, taking us from statements we know or believe are true, to new truth claims.

We speak here in ideal terms. In fact, our reasoning often takes us from one set of falsehoods to another. But how does this happen? How is it possible that our reasoning – if it is good, if it is supposed to lead us to truth – ends up leading us to falsehood? The answer to this question lies in the distinction between an argument’s form or structure and its content. In this section of the course we will look at argument structure and the quality of validity and we will learn how to apply some of the tools logicians use to demonstrate an argument’s structural quality.

To understand deductive validity, we will learn two methods of evaluation developed by philosophers and mathematicians in the late 19th and early 20th century. These methods rely on the symbolization techniques introduced in the first section of the course, namely, the logical languages of Predicate and Sentential. As the primary tool for revealing the structure of deductive arguments, symbolization lets us easily test their validity using one or both of these methods.

One method we will use is truth tables. The truth table method will demonstrate whether or not given Sentential arguments are valid or invalid. This method is a simple extension of truth table techniques previously introduced. A second, more powerful method of deductive proof will show in precise and explicit terms how for any valid argument, a given conclusion can literally be drawn out from a set of premise statements, through a sequence of relatively simple inferences. This method involves using rules of inference that work with the logical meaning of the five logical operators and two quantifiers already introduced.

Deductive Argument Evaluation: Validity and Invalidity

All deductive arguments are either valid or invalid. In the context of argument evaluation, these terms have a narrow, technical meaning. Whether or not an argument is valid or invalid, depends on whether its conclusion can in fact be drawn out of its premises. It does not refer to the actual, factual truth or falsehood of the statements in the argument, and therefore has nothing to do with the specific content of an argument. Validity is a feature of the structure of an argument alone. When we symbolize an argument using a logical language, we expose that structure, bringing it into the foreground where it is easier to test.

The distinction between valid and invalid is mutually exclusive. This means that no deductive argument can be both valid and invalid. If an argument is valid, it cannot possibly be invalid. If it is invalid, it is structurally flawed and the premises fail to provide sufficient support for the conclusion. A valid argument is valid because it will not admit the possibility that a false conclusion can be drawn from a set of true premises. This fact has to do only with the structural relationship that exists within its premises. If the argument is invalid, then it contains a structural flaw such that it is possible for it to have nothing but true premises and yet its conclusion is false. Again, the formal invalidity of an
argument does not involve the argument’s content. The content may be factually accurate, or it may be quite believable. But this is irrelevant. The only concern we will have when we test for validity is whether or not the underlying relationships that link the premises together in fact provide the ground from which the conclusion can be drawn forward.

The fact that validity and invalidity have nothing to do with an argument’s content – either the specific subject matter the argument addresses, or the truth value of its statements, leads to some surprising facts about deductive arguments. For one, an argument can be invalid even when every statement in it is factually true. In such cases, even though the conclusion and premises are true, the conclusion does not follow from the premises. The premises do not in fact provide necessary support for the conclusion, and we cannot derive the conclusion from the premises. It is also possible for a valid argument to contain only false statements. In such cases the conclusion will follow from the premises, based upon the structural relationships among premises. The premises give structural support to the conclusion, but, because the conclusion is false, at least one of the premises must also be false. The point to take away here is that deductive logic is not, strictly speaking, about truth. It is, more fundamentally, about the structural integrity of arguments.

**Valid vs Invalid: Definitions**

We use the terms valid and invalid to distinguish good argument structure from bad. The technical meaning of the term “validity” is contained in the following definition:

**Deductive validity = df:** An argument will be valid if, and only if, given the form of the argument, it is not possible for a set of true premises to yield a false conclusion.

This definition makes clear that validity is a matter of argument form and not content. It also makes clear that validity is a function of the truth value possibilities inherent in a given argument’s form. If we consider all truth value possibilities between any given set of premises and a given conclusion, we find there are only four:

- **true premise set** → **true conclusion**
- **true premise set** → **false conclusion**
- **false premise set** → **true conclusion**
- **false premise set** → **false conclusion**

Only the second possibility (true premise set → false conclusion) reflects the structural flaw that makes a deductive argument invalid. According to the definition of validity, an argument is valid if, and only if it is impossible for this case to exist. If we want to show an argument to be valid, the easiest way to do so, is to test for this specific case – to try and show that the form of the argument has the possibility of having all true premises and a false conclusion. If it is impossible to establish this case, then the argument is necessarily valid.
Demonstrating Validity and Invalidity: Truth Tables

Because validity is a feature of argument form, and because truth tables present a complete account of the truth value possibilities for any statement or sets of statements, we can use truth tables to determine whether any truth-functional deductive argument is necessarily valid or invalid. Sentential is the logical language that represents the truth-functional structure of statements and arguments. So, we will use it to illustrate this method to show (in)validity. Because quantified arguments involve variables, we must adapt this method for arguments containing quantifiers; therefore we will not use this method to arguments translated into Predicate, in this course. In many cases, however, their underlying structure can easily be seen to be analogous to common truth-functional forms.

The truth table method to determine (in) validity, is essentially the same method used previously for statements. As before, you need to know how to set the tables up, how to fill them in, and how to read them. The first two are exactly the same. The difference will be in the reading of the table. You begin as before, establishing guide columns using the argument’s simple statements to determine the number of rows and adding columns until you have accounted for all the truth value possibilities for all statements within the argument. Determining validity or invalidity is a simple matter of comparing values for the premise statement(s) against those of the conclusion.

The table below is repeated from Part I to help you review the procedures for setting up truth tables.

<table>
<thead>
<tr>
<th>TRUTH TABLE REVIEW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. <strong>ESTABLISH GUIDE COLUMNS</strong> by identifying the number of simple statements in the context. This requires counting the number of simple statement symbols.</td>
</tr>
<tr>
<td>2. <strong>ESTABLISH THE REQUIRED NUMBER OF ROWS</strong> by using the formula, $2^n$. This entails that the number of rows will expand exponentially. Beginning with a necessary two rows, (for an argument that contains only one simple statement), tables expand to 4, 8, 16, 32 and so on, with each additional simple statement.</td>
</tr>
<tr>
<td>3. <strong>FILL IN THE GUIDE COLUMNS.</strong> From the far right guide column alternate Ts and Fs for the required number of rows. Double Ts over Fs for each column to the left.</td>
</tr>
<tr>
<td>4. <strong>BUILD COLUMNS TO THE RIGHT.</strong> In addition to guide columns, you need one column for every logical operator in the statement or argument. To determine the total number of columns required use: $#$ of guide columns $+$ $#$ of logical operators</td>
</tr>
<tr>
<td>5. <strong>READ THE TRUTH TABLE.</strong> The specific row(s) or column(s) you focus depends on whether you need to determine the value possibilities for a statement, or some feature about an argument or set of statements. To determine the value possibilities of a statement (tautology, contradiction, logical equivalence, etc), “read” columns. To determine a fact (validity/invalidity; consistency/inconsistency) about an argument, “read” rows.</td>
</tr>
</tbody>
</table>
Let’s consider two simple and closely related argument forms, to show how this works:

A. \( p \lor q \quad \sim p \quad \therefore q \)

B. \( p \lor q \quad p \quad \therefore \sim q \)

These argument forms have obvious similarities. They both contain a disjunction premise. Their second premise addresses the truth value of \( p \), and their conclusions infer something about the truth value of \( q \). Because truth tables allow us to calculate the possible truth values of any truth-functional statement, we can use a truth table to determine whether these argument forms are valid.

For both tables A and B, the first two “guide” columns are identical because the two arguments contain only two simple statements, symbolized as \( p \) and \( q \). These columns generate the truth value combinations for the simple statements in the arguments. Columns three and four represent the premises of each argument and column five, their conclusions. These columns are clearly identified.

Both tables show two possibilities where the conclusion is false. In argument A, rows 2 and 4 show a false conclusion. In argument B rows 1 and 3 show a false conclusion. Now, look carefully at the table for argument A, and note that for each row where the conclusion is false, there is at least one false premise. In row 2 where the conclusion is false, the premise, \( \sim p \), is also false. In row 4 where the conclusion is false, the premise, \( p \lor q \), is false. Argument A is, therefore, valid. Every time “F” appears in the conclusion column, there is at least one “F” under a premise column for that row. Therefore, there is no possibility for argument A to have all true premises and at the same time, false conclusion.

For argument B, however, we need only look at row 1, where the conclusion, \( \sim q \), is false, but both premises are true. This row shows the possibility of having true premises that lead to a false conclusion. Whenever a truth table shows a single row in which all premises are true and the conclusion is false, the argument is invalid. Do not confuse the case of an argument having true premises leading to a true conclusion with validity. This would be fallacious and the two tables above illustrate why this is so. Compare row 3 of table A and row 2 of table B. These rows show all statements in both arguments to be true. While this may seem to indicate a good or “valid” argument, it does not. Argument B is clearly invalid by row 1. Therefore, having all true statements is not a reliable indicator of validity.
Since validity and invalidity are mutually exclusive, any given argument will be valid or invalid. **No argument can be both.** In a truth table, a specific row or rows will show invalidity. If this possibility is not shown – if there is no row in the table where all premises are true and the conclusion is false, then the argument is not invalid. This means the argument is valid.

Truth tables provide a definitive test for formal validity for any deductive argument. While this information is definitive, it is limited. Truth tables do not tell us which of the possibilities (which row) is the actual, factual case. Therefore, the conclusions we can derive from a truth table – while objective – are limited. We can have certainty, but it is a formal, abstract certainty that does not touch upon actual truth value of the specific statements in the argument. Having said this, we should not underestimate the power of deductive certainty; for often surprising results come from exposing the abstract, formal features of arguments.

**Setting up truth tables for arguments: Full Truth Table Method**

Accurate tables require the correct number of rows and columns so that all the required information is generated. To determine how many rows are required remember the simple mathematical expression, \(2^n\), where “2” represents the number of truth values, true and false. The superscript \(n\) is replaced with the number of the simple statements in the context. To find the value possibilities of \(A \lor \sim A\), for example, the superscript \(\text{n}1\) will be replaced with 1. The result will be \(2^1\) and there will be two rows in the truth table. The first row will contain a T, indicating when \(A\) is true, and the second row will contain F, indicating when \(A\) is false. If we have three simple statements in a given context, \(A, B, \text{ and } C\), then the table will require \(2^3\) or eight rows.

To ensure consistency and that no combination of truth values is overlooked, we begin filling in the guide columns (headed by the simple statements in the context) by going to the furthest column to the right, alternating Ts and Fs for the required number of rows. For each column to the left, we double the Ts and Fs from the previous column.

The number of columns depends upon the number and complexity of the statements in the argument. We must account for each premise, and for each operator in the statement – again, always building from simple to complex. From the guide columns we create columns. There is some play with how columns are built but you should account for each logical operator in each statement within the argument, building complexity from left to right. There is no necessary order for the columns. Given the simple statements, \(A, B, C\), one person may choose to order of columns alphabetically, another in the order the statements appear in the context. One person will choose to put negative simple statements directly after the simple statements; another will place them as they appear in a given premise or conclusion statement. This will not affect the final result. If the table has been set up with the necessary number of rows, and if the columns contain all the necessary data required, the table will provide a definitive determination of validity or invalidity.
Reading Truth Tables for Validity and Invalidity

To read a truth table, you need to be clear about what you are looking for. If you want to know if an argument is valid or invalid, you only need to look at the rows in the table where the conclusion is false. It is most efficient to just look first to the conclusion column and locate every row where there is a false conclusion. Then compare the instances of false conclusion to the values for the premise statement(s) in these rows only. You are looking to see if there is one row where there are only true premises at the same time that the conclusion for that row is false. If at least one row in which the conclusion is false while every premise in that row shows a T, the argument is invalid. However, if there is a false premise in every row where the conclusion is false, the argument is valid.

REMEMBER: An argument is invalid if, and only if it is POSSIBLE for the entire premise set to be true while the conclusion is false. It is valid only when it is IMPOSSIBLE for all its premises to be true and its conclusion to be false. If an argument is invalid, it is possible to identify the specific row(s) that show where all premises are true while the conclusion is false. If an argument is valid, there will be no such row. The table as a whole illustrates validity.

Truth tables provide a purely mechanical means for determining the validity or invalidity of a given deductive argument. If the number of columns is manageable, a truth table offers a simple technique that gives us a definitive answer on the question of validity. Yet, for each additional simple statement added to the argument, the rows expand exponentially, and columns expand with the number and complexity of premises. With each additional row or column, the opportunity for error also increases. So, it does not take long before truth tables become too large to make them a useful tool. They quickly become tedious, time consuming and inefficient. Fortunately, there are short-cut techniques can be used to cut the time and effort required.

This said, it is important to practice. Daniel Willingham, a cognitive psychologist makes it clear: "It's virtually impossible to become proficient at a mental task without extensive practice . . . if you repeat the same task again and again, it will eventually become automatic. Your brain will literally change so that you can complete the task without thinking about it." If you wish to be successful in the techniques presented in this course, then you must find the time to practice.

Truth Table Reading Exercise: Read the tables below to determine if the arguments represented are valid or invalid. Premise and conclusion columns are labeled. If invalid, clearly and explicitly identify at least one specific row that shows this by stating: Invalid by row #.

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>P1 p = q</th>
<th>P2 p v q</th>
<th>P3 ~p</th>
<th>∴ ~q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>
2. 

<table>
<thead>
<tr>
<th>p</th>
<th>~p</th>
<th>P1 p v ~p</th>
<th>⇒ p ⊃ ~p</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

3. 

<table>
<thead>
<tr>
<th>p</th>
<th>~p</th>
<th>P1 p . ~p</th>
<th>⇒ p ⊃ ~p</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

4. 

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>~q</th>
<th>P1 p . ~q</th>
<th>P2 ~q ⊃ p</th>
<th>~p</th>
<th>⇒ ~p ⊃ ~q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

5. 

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>r</th>
<th>~q</th>
<th>P1 q . ~q</th>
<th>⇒ p ≡ r</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
**Review exercise – Filling in truth tables:** For the following tables, fill in all values. Then, read the table to determine whether the argument indicated is valid or invalid.

6.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>P1 \ p v r</th>
<th>P2 \ q = r</th>
<th>P3 \ p = q</th>
<th>\sim ~r</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

7.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>\sim q</th>
<th>\sim p</th>
<th>P1 \sim p v q</th>
<th>P2 \sim q \Rightarrow \sim p</th>
<th>P3 \ p = q</th>
<th>\sim p</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

8.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>\sim p</th>
<th>P1 \ p v q</th>
<th>P2 \sim p v r</th>
<th>\Delta \ p v \sim p</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Exercise:** Practice using truth tables for basic argument forms. Construct truth tables to determine whether the following argument patterns are valid or invalid. Pay close attention to the subtle differences in each argument pattern. These are the differences that distinguish valid from invalid reasoning. (Answers to the odd exercises can be found beginning on p. 78.)

<table>
<thead>
<tr>
<th></th>
<th>1. ( p \implies q )</th>
<th>2. ( p \implies q )</th>
<th>3. ( p \implies q )</th>
<th>4. ( p \implies q )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( p )</td>
<td>( q )</td>
<td>( \therefore p )</td>
<td>( \therefore q )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>5. ( p \lor q )</th>
<th>6. ( p \lor q )</th>
<th>7. ( p \lor q )</th>
<th>8. ( p )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \neg p )</td>
<td>( p )</td>
<td>( \therefore q )</td>
<td>( \therefore p )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>9. ( p \land q )</th>
<th>10. ( p \land \neg p )</th>
<th>11. ( p \land \neg p )</th>
<th>12. ( p \land q )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \therefore p )</td>
<td>( \therefore \neg p )</td>
<td>( \therefore q )</td>
<td>( \therefore p \land q )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>13. ( \neg p \sim q )</th>
<th>14. ( \neg p \sim q )</th>
<th>15. ( p \lor q )</th>
<th>16. ( p \lor q )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \therefore \neg q )</td>
<td>( \therefore q )</td>
<td>( \therefore p \land q )</td>
<td>( \therefore q )</td>
</tr>
</tbody>
</table>

COMPARE the valid and invalid patterns in each set. The valid patterns 1, 8, 9, 12, 14 and 15 will be used as RULES OF INFEERENCE. The other valid patterns can be proven using these basic patterns. Invalid patterns show common mistakes in reasoning that often are made when doing a formal proof. These errors are formal fallacies and are considered serious logical errors. According to the principle of formal analogy, any argument that shares a valid form will be valid, and any argument that shares an invalid form will be invalid or formally fallacious. Studying the shape or form of basic patterns can be useful, because it can help us to recognize whether more complex arguments are valid or invalid.

There are two modifications of the full truth table method that provide short cuts of sorts. These will be presented and then you will have a choice of which method you prefer to use when testing arguments to determine if they are valid or invalid.
Tautology Test: A Second Truth Table Method

You may have noticed a relationship between conditional statements (material implications) and arguments (inferences), or between an “if-then” relationship and an argument having the form: “A is true because B is true.” It is true that any argument can be represented as a conditional statement, in which the premise statements serve as the antecedent and the conclusion serves as the consequent.

Just as a conditional statement is only false when a true antecedent leads to a false consequent, so an argument is only invalid when it is possible for true premises lead to a false conclusion. We use this parallel in a second method to test for validity. To use this method, you convert the argument to a conditional statement. How we do this is made clear by the following example.

EXAMPLE 1: Consider the argument:

I must take either English or Anthropology. I can’t take Anthropology. So, I’ll take English.

This simple argument has two premises and one conclusion, which are easily identified. Remember that premises are statements that are presented as if they were true – although it may be that they are in fact false. Remember also that in a valid argument, the conclusion is claimed to follow from the premises. In other words, its truth is grounded in the truth of the premises. So, we could present this argument as a conditional statement in which the premises occupy the position of the antecedent and the conclusion occupies the position of consequent. For the argument above we would get the following conditional statement:

If I must take either English or Anthropology, but I can’t take Anthropology, then I’ll take English.

Any argument can be expressed as a conditional statement. The corresponding statement will be a conditional statement of the form: If the conjunction of the premises are true, then the conclusion is true. Any such statement, will present a valid argument if, and only if its corresponding conditional statement is a tautology.

This “tautology test” is useful with simple argument forms of no more than three simple statements. With arguments more involved than this, you will encounter the same problems as with the full truth table method.

REVIEW AND EXAMPLE 2: To apply the method use the following steps:

Step 1: Create a conditional statement in which the premises form the antecedent, and the conclusion the consequent. If there is more than one premise then create a conjunction statement linking the premises to form the antecedent.
Argument:     Conditional Statement:  
A v B        becomes    [ (A v B) • ~ A ] ⊃ B
~ A     
∴ B

**Step 2:** Take [ (A v B) • ~ A ] ⊃ B, and develop a truth table to determine if this statement is a tautology. Use the “shortcut” method for statements to determine whether the statement were tautologous, contingent or contradictory.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>[ ( A v B) • ~A ]</td>
<td>⊃</td>
<td>B</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

To read the table, consider only the column for the main conditional operator, – which will always be the conditional linking the premise to the conclusion. In this example, the table shows **the conditional statement is a tautology**. This shows that the original argument is valid.

Another EXAMPLE:

Argument:     Conditional Statement:  
A ⊃ B        becomes    [ (A ⊃ B) • ~ A ] ⊃ ~B
~ A     
∴ ~ B

Now YOU establish a truth table to see if this conditional statement is a tautology.

**Exercise:** Apply the tautology test to the following arguments to determine their validity. 
*answers on p. 79-80.

1. A ⊃ B  
   ~ B   /∴ A

2. A ≡ B  
   B v ~ A /∴ A v ~ B

3. B ≡ ~ A  
   ~ B   /∴ A

4. C • B  
   B ⊃ A   /∴ C • A

5. A v B  
   B ⊃ C   /∴ ~ A

6. C ⊃ A  
   C v B   /∴ A v ~ B
A Super Short-Cut Method

You now should know how to use truth tables to determine whether or not any given truth-functional argument is valid. You should know 1) how to set them up, 2) how to fill them in and 3) how to read them for validity or invalidity. They present a relatively simple algorithm or set procedure that can be applied to any truth-functional deductive argument. While they are simple, they can be a pain. Where the number of simple statements exceeds four (which means you would need to construct a table with at least 32 rows) they are tedious and justifiably feel like a waste of time.

But, if you understand how they work, you can determine (in)validity using a third, “super short-cut” method. All you have to do is identify the truth value assignments for each simple statement in the argument that correspond to any row in a full truth table that shows invalidity.

To use this method, you will need to solidly and confidently know two things:

1. The definition of invalidity
2. How to determine what values will make any given statement true or false – given the logical operators that structure the statement.

With this knowledge, follow 3 simple steps:

**Step 1: FORCE the conclusion to be False.**
Assign truth values to the separate simple statement in the conclusion that will make the conclusion FALSE.

**Step 2: TRY to force all the premise\statement(s) to be TRUE.**
Take the truth values assigned to the simple statements in the conclusion and assign these same values to the corresponding simple statements contained in the premises. Then find truth values for the remaining statement statements that will make each premise TRUE.

Make sure that you are consistent in assigning only one value (T or F) to any given simple statement throughout the entire argument. (This means that if you give the statement A, a true value (T), every occurrence of A in the argument must be assigned T. If you assign false to B, then every occurrence of B must be F.)

Steps 1 and 2 are not strictly mechanical. You must contrive the values for every statement in the argument to achieve the result you want. There may be more than one possible assignment of values that will achieve this result (False conclusion and all true premises).

**Step 3: Determine validity or invalidity**

If you can make all the premises true while at the same time making the conclusion false, you have shown the argument is INvalid. If you cannot do this, if every assignment of truth values that makes the conclusion false, makes one or more of the premises false, then the argument must be valid.

This same result will be shown if every time you make one premise true, it necessarily makes another false, then you have a valid argument. The reason for this result is that it
is impossible to make the premise set true, when the conclusion is clearly been set to be false.

This “super short cut” method is really just trial and error (or guess and check). There is no magic in it. You have to begin with some assignment of truth value. After you have one or two values for individual simple statements, you just work through the argument like a puzzle trying to figure out how to get all the premise statements to have a true truth value, while the conclusion value stays false.

The type and complexity of the statements you are dealing with, and the truth values you need to contrive, limit the possibilities, so the best strategy is to always start with the statement that has the most limited possibilities for being true or being false. For example, if the conclusion is a simple disjunction, there is only one possible assignment of values that will make it false. If it is a conjunction, however, three possible combinations will make it false. To begin, you just have to choose one of them and see if it works. If not, try again until you can make a necessary judgment, or you have exhausted all possibilities.

Use the “substitute and calculate” method to check your accuracy. The “substitute and calculate” method is a transparent and self-corrective way of ensuring a correct answer. Always check to make sure the truth value assignments you have given to simple statements work to achieve the goal of having all true premises and a false conclusion.

While you can use this “super short-cut” method for any argument, it becomes a real advantage for arguments with three or more simple statements when the other methods would require too many columns and rows. It also works best when the conclusion is either a disjunction or a conditional statement because there is only one possibility for such statements for being false, thus making it relatively easy to assign the necessary value. Having a premise that is a conjunction also significantly narrows the possibilities one has to consider.

Two examples will show how to apply this method. The final answer is presented using a short, informal argument.

Example 1:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Goal &amp; check</th>
<th>grid keeps track of value assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>M v T</td>
<td>F v T = T</td>
<td></td>
</tr>
<tr>
<td>~ T</td>
<td>~ F = T</td>
<td></td>
</tr>
<tr>
<td>∴ M</td>
<td>F = F</td>
<td></td>
</tr>
</tbody>
</table>

Answer: Any truth value assignment that makes the conclusion false and premise 1 true, makes premise 2 false. Any assignment that makes the conclusion false and premise 2 true makes premise 1 false. Therefore, it is not possible to make the premise set true while the conclusion is false. Therefore, the argument is valid.

Example 2:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Check and goal</th>
<th>grid keeps track of value assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>D ⊃ C</td>
<td>F ⊃ F = T</td>
<td></td>
</tr>
<tr>
<td>~ C</td>
<td>~ F = T</td>
<td></td>
</tr>
<tr>
<td>∴ L v D</td>
<td>F v F = F</td>
<td></td>
</tr>
</tbody>
</table>


Answer: The grid shows a possible assignment of truth values to all simple statements in the argument that will show all premises true and the conclusion false. Therefore, the argument is invalid.

Super short Cut Method Exercise: Practice using the short cut method to show the validity or invalidity of the following arguments. Use the two examples above as models to check your work and make sure it is presented clearly, completely and unambiguously. Answers to * on p. 80.

*1. \[ P \equiv Q \]
\[ \sim Q \lor R \]
\[ \therefore R \supset \sim P \]

\[
\begin{array}{ccc}
P & Q & R \\
\hline
\end{array}
\]

2. \[ S \lor (\sim T \lor U) \]
\[ \sim U \supset \sim T \]
\[ (S \land U) \equiv T \]
\[ \therefore \sim U \lor T \]

\[
\begin{array}{ccc}
S & T & U \\
\hline
\end{array}
\]

*3. \[ \sim (H \cdot \sim I) \]
\[ I \lor (J \equiv K) \]
\[ K \cdot \sim H \]
\[ \therefore \sim (J \cdot I) \]

\[
\begin{array}{ccc}
H & I & J & K \\
\hline
\end{array}
\]

4. \[ Z \supset \sim \sim T \]
\[ \sim (M \supset \sim T) \]
\[ F \supset Z \]
\[ M \supset \sim G \]
\[ \therefore G \supset \sim F \]

\[
\begin{array}{cccc}
Z & T & M & F & G \\
\hline
\end{array}
\]

5. \[ A \cdot (B \cdot C) \]
\[ \sim (\sim C \lor \sim D) \]
\[ A \equiv (D \equiv C) \]
\[ \therefore C \lor B \]

\[
\begin{array}{cccc}
A & B & C & D \\
\hline
\end{array}
\]

6. \[ \sim (N \lor O) \]
\[ \sim S \supset \sim Q \]
\[ O \supset Q \]
\[ \therefore \sim (N \cdot S) \]

\[
\begin{array}{cccc}
N & O & S & Q \\
\hline
\end{array}
\]
7. \[ F \equiv (B \lor U) \]
   \[ \sim B \equiv (U \lor N) \]
   \[ M \cdot \sim N \]
   \[ E \vdash \sim M \]
   \[ E \lor F \]
   \[ \therefore N \]

8. \[ A \lor (T \lor U) \]
   \[ L \Rightarrow T \]
   \[ (A \cdot U) \equiv G \]
   \[ \therefore G \cdot T \]

9. \[ \sim (H \cdot X) \]
   \[ X \equiv (O \equiv R) \]
   \[ R \cdot \sim H \]
   \[ \therefore O \vdash \sim X \]

10. \[ A \vdash B \]
    \[ \sim C \vdash \sim B \]
    \[ F \vdash A \]
    \[ C \cdot \sim C \]
    \[ \therefore \sim A \vdash \sim F \]

11. \[ M \cdot (T \cdot U) \]
    \[ \sim \sim U \vdash \sim M \]
    \[ T \equiv (L \equiv T) \]
    \[ \therefore L \lor T \]

12. \[ Q \lor R \]
    \[ \sim S \vdash \sim Q \]
    \[ R \vdash W \]
    \[ \therefore S \lor W \]
FOCUS ON CONTENT: Reading truth tables for consistency

If thought or intelligence is the means of intentional reconstruction of experience, then logic, as an account of the procedure of thought, is not purely formal. It is not confined to laws of formally correct reasoning apart from truth of subject matter.

John Dewey. Reconstruction in Philosophy

From a strictly logical perspective the structural integrity of the argument is key. If there is a flaw at this level, the argument has no logical credibility. But this does not mean that all valid arguments are good arguments. They are only good from the perspective of structure or form. Even if an argument is assessed as valid, there are other ways in which its quality can be compromised.

If a deductive argument is valid, its structure is solid. If its premises are true you can be certain that its conclusion will be true. But many valid deductive arguments have premises that are false. It is a fact that to assess the truth value of statements in any given argument, you must have specific knowledge of the particular subject matter that is being argued. Logicians have no special authority here. However, since, truth tables cover all possible situations, they can give some information relevant to the issue of argument soundness. In this context, the concepts of consistency and inconsistency are important.

Consistency/Inconsistency

The terms consistent and inconsistent refer to whether or not a set of statements has the possibility of being all true (consistent) or not (inconsistent). If a set of statements is consistent then the truth table for that set of statements will show at least one row in which every statement is true (T). If a set of statements in inconsistent then the table will now show such a row. Every row will have at least one F for at least one of the statements within the set.

Since you already know how to set up truth tables for sets of statements, all that is required to determine whether or not the set is consistent or inconsistent is to “read” the table for this possibility. If you determine that an argument is both valid and consistent, then the most you can say is that it has a possibility of being sound. If it is valid but inconsistent, it is definitely unsound. And any argument is invalid, it is automatically unsound.

EXAMPLE 1: Consider the argument in following table.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>P1</th>
<th>P2 B ⊃ A</th>
<th>P3 A v ~B</th>
<th>⊼ ~A</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

The table shows the argument invalid by row 2 (where all premises are true and the conclusion is false). Therefore, it is automatically unsound. It is, however, consistent by row 4 because in this row all statements are true, which means it is possible for the argument to have all true statements. In such a
case, even though it is consistent, this argument is unsound because it is invalid.

EXAMPLE 2:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>P1 ~B</th>
<th>P2 A ⊃ B</th>
<th>∴ ~A</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

This table shows the argument to be valid because there is no row where premises are all true and the conclusion false. It is also consistent, by row 4. Can you say it is sound? No. You cannot say definitively that it is sound, because soundness depends on the actual, factual truth of the statements in the argument. Given the symbolic representation are only assessing the argument from the perspective of structure. You have no specific knowledge of the actual statements represented by A and B, much less their actual truth value. What you can say is there is a possibility that this argument is sound.

Reading Exercise: Read each table below to determine consistency and inconsistency. What judgments can you make about the soundness of any of the arguments represented in these tables. What can be said of their soundness?

1.

<table>
<thead>
<tr>
<th>G</th>
<th>~G</th>
<th>P1 G ⊃ ~G</th>
<th>∴ G v ~G</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

2.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>~A</th>
<th>~B</th>
<th>P1 A ⊃ ~B</th>
<th>P2 B v A</th>
<th>P3 ~ A ⊃ B</th>
<th>∴ A • ~B</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>
3.

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>~T</th>
<th>P1 L ≡ ~T</th>
<th>P2 ~T ⊃ L</th>
<th>P3 L v ~T</th>
<th>∴ ~ (L ≡ ~T)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

4.

<table>
<thead>
<tr>
<th>P</th>
<th>Q</th>
<th>R</th>
<th>~R</th>
<th>P1 Q v ~R</th>
<th>P2 Q = R</th>
<th>P3 P ⊃ Q</th>
<th>~P</th>
<th>∴ ~R v ~P</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

**Exercise: Truth Table Practice:** Fill in the truth tables below. You will need to supply intermediate columns. Then read the table to determine validity/invalidity and consistency/ inconsistency. Be sure to reference your answers to specific data in the tables.

1.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>P1 B v A</th>
<th>P2 ~(B • ~A)</th>
<th>∴ A</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2.

<table>
<thead>
<tr>
<th>A</th>
<th>~A</th>
<th>P1 A v ~A</th>
<th>P2 ~(A ⊃ ~A)</th>
<th>∴ ~ ~A</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise: Develop Truth Tables to determine Consistency or Inconsistency for the following arguments. What does each table tell you about the soundness of the argument? You may try to use either a full truth table or the super-short cut method. (Answers to odds p. 82)

1. $T \lor R$, $T \rightarrow \sim R$, $R \rightarrow \sim T$
2. $(S \equiv T) \equiv \sim S$
3. $A \rightarrow \sim A$
4. $\sim P \equiv (Q \lor R)$, $\sim R \rightarrow P$
5. $A \rightarrow (B \cdot C)$, $\sim B \rightarrow A$
6. $A \rightarrow B$, $(\sim A \cdot B)$
7. $A \lor D$, $B \lor C$, $A \rightarrow \sim B$
8. $(V \lor D) \cdot J$, $D \rightarrow J$, $J \equiv \sim V$

**Truth Tables for Evaluating Arguments:**

Truth tables are a perfectly adequate device for evaluating several important features of deductive arguments. They can definitively determine whether or not a given truth-functional argument is valid or invalid, and consistent or inconsistent. They are a simple, strictly mechanical device. There is a set procedure for setting them up and filling them in. You read them for specific information that is defined by definitions that are unambiguous. These are their benefits. Their liability comes with the fact that they are tedious to use, for some they are just boring. It is also easy to make a simple mistake that will cause you to get the answer. Also, for compound statements or arguments that have three or four simple statements, they just get too long and time consuming. Finally, while truth tables are an adequate demonstration for validity, they do not show precisely why this is the case, and this is why logicians use the deductive proof to show exactly how a valid argument is valid.
FORMAL DEDUCTIVE PROOF: THE GOLD STANDARD

A simple person believes every word he hears;
A clever one understands the need for proof.

Proverbs 14:15

In formal Logic a deductive proof is the “gold standard” of demonstration. It displays all the values of deductive logic. A formal deductive proof is a precise, explicit, clear, transparent, thoroughly justified demonstration that allows us to draw an argument’s conclusion from its premises, step by step, through a series of simple valid inferences. A proof is not only a convincing indication that an argument is valid, it shows exactly how the conclusion can be pulled out, or derived from the premises.

Here it is important to remember that one of the features of a deductive argument is that the conclusion statement can be found somewhere within the premise statements. In a proof, you actively draw out that conclusion, by using your own ability to reason deductively by drawing conclusions from a given set of premises using only a prescribed set of rules. The premises and the rules set the boundaries within which you literally pull the conclusion forward, step by step. If you do this correctly you will have made clear the underlying reasoning that links the premises directly to the conclusion.

Every valid deductive argument has a proof (perhaps more than one) that will show how the conclusion can be derived. But an invalid argument has no proof, nor can a proof show that an argument is invalid. Not being able to successfully develop a proof, also does not mean that an argument is invalid. Invalidity means that it is not possible to derive the conclusion from the premises given. If an argument is invalid, it has no proof.

We only use proofs for valid arguments. Completing a proof is not as simple and direct as building a truth table. To “do” proofs you need to have 1) basic knowledge of the conventions for setting up and working through a proof, 2) understanding of how the rules of inference are used, and 3) a willingness to play with and develop skill and strategies. In addition, you need tenacity and a willingness to try something, throw it out and begin again if things are not working. As with any skill, learning to do proofs successfully requires practice. Practice exercises will help you to see how a given rule can be applied in increasingly complex contexts. In this section of the course you will be given only valid arguments.

A proof begins by presenting the argument that is to be proven valid. Premise and conclusion statements are clearly stated. The premises are considered true for the sake of argument (although we know they may not be true). The conclusion is clearly indicated by the conclusion symbol. Premises are the starting point and you use them to derive the conclusions using rules that are valid forms of inference. After it is justified, you can use each successive conclusion you draw forward as premise for further deductions. All derivations are justified in two ways: 1) by reference to a valid rule of inference, and 2) by reference to one or more statements that serve as the premise from which the conclusion is derived.

This section of the course will introduce a standard model of proof – the two-column proof, and a basic set of rules of inference. The approach will be to introduce new rules and exercises that use these rules, giving you a chance to understand how they work. New rules will be given, a few at a time, until the complete set of rules has been presented. With each new rule, the range of derivations that can be justified and the sophistication of reasoning strategies you will have available will increase, but the basic approach will always be the same.
RULES OF INFERENCE

Every step in a deductive proof involves making a valid inference or drawing a valid conclusion from previous statements in the proof. Drawing conclusions that match the patterns of rules of inference ensure each step of the proof is a valid inference. There are two basic types of inference rules: Quantifier rules and Operator rules. Quantifier Rules tell you how to manipulate the universal (\(\forall x\)) and existential (\(\exists x\)) quantifiers of predicate. Operator Rules focus upon the logical operators, showing when and how you can derive conclusions from each logical operator, and when and how operators can be introduced.

Each rule will be illustrated by a schematic, or blueprint that shows how the rule works. Exercises will allow you to practice how to use rules correctly in the context of formal deductive proofs. We begin with Quantifier rules.

QUANTIFIER RULES

All quantifier rules are immediate inferences. This means you can draw a conclusion directly, immediately, from only one premise. All quantifier rules manipulate quantifiers. Quantifiers Elimination rules let you take off a quantifier. Because quantifiers coordinate with variables, you must change the variable to an individual constant at the same time. Quantifier Introduction rules let you introduce a quantifier or put a quantifier on a statement that speaks about an individual. You must reintroduce the variable at the same time. The variables must coordinate with the quantifier. The Logical Equivalence rule for quantifiers will let you change a quantifier, but only if you also move a negation sign. Some of these rules have restrictions and you need to pay carefully attention to these. We’ll start with the rules that have no restrictions.

Universal Elimination rule:

Universal Elimination rule (abbreviated as \(\forall E\)), can be applied to any universally quantified statement. Such a statement claims that “everyone” or “all” members of some class (or situation) have a given property, quality or relation. The rule simply expresses the common sense judgement that if a feature is true of all members of a class, then it must be true of each and every individual within the class. Therefore, if I state, “All full time students carry twelve credit hours,” I can conclude directly from this statement alone that if any specific individual (Sally, or Barry, or Tito) is a full time student, then he/she carries twelve credit hours. This is what is expressed in the schema that shows the rule.

Universal Elimination Rule (\(\forall E\)):

\[
(\forall x) \varphi x \\
\therefore \varphi a
\]

where “\(a\)” is any individual constant.

Presented in this way, the rule shows the general pattern for the rule. The first line states: \((\forall x) \varphi x\). This refers to some premise or derived statement in your argument. It means “for any universally quantified statement whatsoever.” The symbol \(\varphi\) (phi) represents any predicate term. In an actual symbolized argument, \(\varphi\) is replaced a predicate symbol (A–Z) such as \((\forall x) Qx\), \((\forall x) \sim Ax\), or by a relationship as we find in categorical or other compound relations, such as \((\forall x)(Ax \supset Bx)\), \((\forall x)(Wx \lor \sim Sx)\), etc.
The schema of the rule shows that when you apply the rule and move from any universally quantified statement to a valid conclusion, which is indicated by the three-dot triangle symbol (ˆ). It also shows that you must do two things: 1) drop or remove (eliminate) the quantifier, and 2) replace the variable (x) with an individual constant (a-w). In the schema the individual constant is represented by α (alpha).

Universal Elimination has no restrictions. You can use it whenever, whereever you see a universal quantifier. It can be used on a whole statement such as (∀x)(Ax ⇒ Bx), or on a part of a statement such as (∀x)¬Ax v (∃x)Cx. You can use it on the same statement as many times as you want. So, for example, if everyone is a student, (∀x)Sx, you can conclude that John is a student (Sj), and Sue is a student (Ss), and Elena is a student (Se), etc.

As a second example, if I claim, “All humans are mortal,” (∀x)(Hx ⇒ Mx), I can validly conclude, from this statement alone, that “If Susan is human, then Susan is mortal,” and “If Mary is human, then Mary is mortal,” and “If Socrates is human, then Socrates is mortal,” and “If Dick Cheney is human then Dick Cheney is mortal,” and so on. I can draw an infinite number of such conclusions validly from this single statement, substituting different named individuals for the variable x.

Existential Introduction rule (∃I):

Existential Introduction rule (∃I) also has no restrictions. The rule states that whenever you have an individual constant that is said to have some characteristic or quality, you can validly conclude that “someone,” or “some x” has that quality. Again, this rule is a simple, intuitive inference that follows immediately from the premise given. For example, if I claim that “George is a student” (Sg), then it follows that “Someone is a student,” (∃x)Sx. If I claim, “Tatiana is not happy” (¬Ht), it follows that “Someone is not happy,” (∃x)¬Hx.

The rule is shown in the same way, using ϕ to refer to any predicate term, and alpha (α) to stand for individual constants. When you use this rule, you take a statement about an individual and you do two things: 1) you add the Existential quantifier (∃x), and 2) you replace the individual constant with a variable.

Existential Introduction Rule (∃I):

```
ϕα
∴ (∃x) ϕx
```

The issue of bound and free variables

Variables are placeholders. They can be bound or they can be free. A bound variable is bound to a quantifier or lies within the scope of a quantifier. Parentheses and brackets are used to indicate the scope of the quantifier. Any variable not within the scope of a quantifier is considered free. No expression containing a free variable is a statement because it does not have truth value. For example, in any predicate function, such as Ax, the variable is not bound by a quantifier. It has meaning – which we can find in a language key, but it does not have truth value.

As a matter of convention any expression that stands directly to the right of a quantifier is bound
by or within the scope of the quantifier. We indicate how far the scope or range of a quantifier extends by using parentheses and brackets. When you use a quantifier Elimination rule, you must replace all bounds variable with the same individual constant. Likewise, when you use quantifier Introduction rules, you must indicate clearly what variables are bound to the quantifier by using parentheses or brackets.

**Example 1:** $$(\forall x) Ax$$

“$$(\forall x) Ax$$” a one-place predicate truth claim. It means “Everything is a A.” It is understood that the variable “x” is bound by the quantifier because Ax stands directly to the right of the quantifier. You can use Universal Elimination rule to conclude statements such as Aa, Ab, Ac, Ad, and so on.

**Example 2:** $$(\forall x) (Ax \supset \neg Fx)$$

In this example parentheses indicate that the two instances of the variable “x” are bound by the quantifier. Using the Universal Elimination rule require that you replace both variables with the same individual constant. This would allow derivations such as Aa $\supset \neg Fa$, Ab $\supset \neg Fb$, Ac $\supset \neg Fc$, and so forth.

**Example 3:** $$(\forall x) Ax \supset \neg Fx$$

Note: This is not a well-formed formula but shows the common error of not using parentheses properly.

In this example, the lack of parentheses means that only Ax is bound by the quantifier. When you apply the Universal Elimination rule, you can only replace the first “x.” This would allow us to conclude Aa $\supset \neg Fx$, Ab $\supset \neg Fx$, Ac $\supset \neg Fx$, etc., but NOT Aa $\supset \neg Fa$, Ab $\supset \neg Fb$, Ac $\supset \neg Fc$, etc.

Predicate is a logically precise language. The symbols and their order have exact meanings. Small things – such as the placement of a negation sign, or the placement of parentheses – result in completely different meanings. Always pay attention to details and note how details translate into meaning. Consider two examples using Existential quantifiers.

**Example 4:** $$(\exists x) (Sx \cdot Tx)$$

**Example 5:** $$(\exists x) Sx \cdot (\exists x) Tx.$$  

In example 4, the existential quantifier, $$(\exists x),$$ has scope over, or binds both Sx and Tx. The statement can be translated as “Someone is both a student and a teacher,” or as the standard form categorical statement, “Some students are teachers.” When you use Existential Elimination rule, you must replace both the x’s with the same individual constant. In the case of Existential Elimination, we will adopt the convention of using “t” to represent a temporarily identified or named individual.

Example 5 shows two quantifiers, each governing a separate predicate function. The statement means: Someone is a student, and someone (else) is a teacher. In this example you cannot know, and therefore cannot infer, that the statement is making a claim about some one individual. For all you know, “x” in each case is holding the place for two different individuals. In this case, each quantifier must be removed separately-- on a separate line, and each variable must be replaced with a different individual constant, say $t_1$ and $t_2$.

When you introduce a quantifier you need to know which individuals can, or are being bound to it. For example, you may want to generalize the following statement using an existential quantifier: “John is a student and Mable is a teacher.” But what quantified expression will result? What is it logical or reasonable to infer?
Self Practice: Use Universal Elimination \( (\forall E) \) OR Existential Introduction \( (\exists I) \) as appropriate.

Odd answers p. 80. What TWO things must you do when you apply these rules?

| 1. \((\forall x) Qx\)  | 11. \((\forall x) Gx \Rightarrow Gi\) |
| 2. \((\forall x) \sim Kx\) | 12. \((\exists x) Bx \Rightarrow (\forall x) Bx\) |
| 3. \((\forall x) (Sx \lor Tx)\) | 13. \((\forall x) Bx \Rightarrow (\exists x) Bx\) |
| 4. \((\forall x) Mx \cdot Ct\) | 14. \(Bi \Rightarrow (\forall x) Bx\) |
| 5. \((\forall x) (Rx \Rightarrow \sim Fx)\) | 15. \(\sim Jc\) |
| 6. \(Bc\) | 16. \(~ \sim Jc\) |
| 7. \(Ml \cdot \sim Tl\) | 17. \(\sim Je \cdot \sim Mc\) |
| 8. \(Ss \lor \sim Fs\) | 18. \((\forall x) (~ Bx \equiv \sim Cx)\) |
| 9. \(Ss \lor Fl\) | 19. \((Gb \cdot Lb) \lor \sim Kb\) |
| 10. \(Pb \cdot Hm\) | 20. \((\forall x) [Gx \Rightarrow (\sim Fx \sim Cx)]\) |

**Quantifier Rules with Restrictions**

The next two quantifier rules have restrictions. The restrictions limit the situations in which you can use these rules, and require that you be mindful of the logic behind the rule so that we avoid fallacies.

We can always validly reason that since something is true of everyone, it must necessarily be true of someone. In a proof, however, while we can always move from a universally quantified statement to an existentially quantified statement; we must go through an individual constant using \( \exists I \) first, and then use \( \forall E \) rule to introduce the existential quantifier. This is to show the details of the logical links that connect such statements.

What do you think about the validity of moving from an existentially quantified statement to a universally quantified one? When you generalize universally, you claim that a quality or relationship holds for any and all members indicated. We hear such generalizations and use them ourselves in our everyday discourse, but if think carefully, these kinds of claims are rarely true. It is easy to think of an exception to almost any universal or global generalization, unless it expresses a definition, such as all bachelors are unmarried men. We can also limit the universe of discourse and argue about all the students in this room, for example. These limits do not diminish the power generalizing plays in our reasoning, but it points out there are limits to generalizing in a logically valid way.

If we are going to draw a universally generalized conclusion, we cannot validly do so from a specific or named individual. Knowing John is a student leads to the valid conclusion “someone is a student,” but it does not validly lead us to “everyone is a student.” This latter inference is a fallacious form of reasoning known as Hasty Generalization. Hasty Generalization is a fallacy that involves moving from a particular individual or a small group of individual to make a judgment about an entire class. For example, it is not legitimate to reason that just because Jon is a nice guy, that everyone is a nice guy, or that because Alex owns a Mercedes, that everyone owns a Mercedes, or that because Mohamed Atta is a violent extremist Muslim, that all Muslims are violent extremists. We call this instance of Hasty Generalization, stereotyping instance.

In formal proofs we show our awareness of this type of fallacy by using the convention of arguing from a randomly selected individual – one that has the relevant characteristics we are arguing about, and who can represent all members of a class. What is true of an individual selected randomly, is true of individual in the class from which the individual was selected. In other words, it is valid to draw a universally generalized conclusion in such a case.
Science uses this idea of random selection to draw broadly generalized conclusions about a given subject matter. You are familiar with the “randomly selected individual,” as John or Jane Doe, who represent all of us but is nobody in particular. Because we can’t possibly talk about everyone, we focus our reasoning on this figure, “John Doe,” is a model for all of us. We contrast randomly selected individuals to specified, named or otherwise designated individuals, who are referred to by their proper name, or a pronoun, or by describing them in some specific way.

Random selection is also used when we study the effects of a given drug in a controlled experiment. Scientists assign subjects randomly to different groups and then generalize the results. When scientists want to know the degree to which the reef fish population has diminished off a particular coast line, for example, they may section off randomly selected areas of the reef and study the fish population in these sections over time. They then extrapolate the results, drawing conclusions about the health of the reef fish population in general.

**Universal Introduction Rule (∀I):**

Universal Introduction rule illustrates the use of a “randomly selected” individual by restricting the circumstances under which it is valid to draw a universally quantified conclusion. It states that you can validate a universal generalization only when you universalize from a “randomly selected” individual. We will adopt the convention of using the symbol “r” to indicate that we are reasoning from a randomly selected individual. This symbol “r” will work just like any other individual constant, but it will be a special type of individual constant, having the meaning of a randomly selected individual.

Using the ∀I rule follows the basic pattern of the ∃I rule. You must add the quantifier, and replace “r” with the appropriate variable. ∀I rule is presented in the following schema:

<table>
<thead>
<tr>
<th>Universal Introduction Rule (∀I):</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \forall r ) ( \phi r )</td>
</tr>
</tbody>
</table>
| \( \therefore (\forall x) \phi x \) | **Restriction**: where “r” represents a randomly selected individual.

The restriction states you can only validly introduce a universal quantifier to statements that contain “r.” Every “r” you include in the scope of the quantifier will be replaced with a variable, and only “r’s” can be replaced with a variable. In this way you avoid Hasty Generalizations, a fallacy that draws a universal conclusion from a specific individual. In the context of a proof, you must look to the conclusion you want to prove. If it is universally quantified, you will need to go into the proof introducing the “r” as the individual constant.

**NOTE**: We recognize that “r” can be used to specify a named individual, but this must be indicated by a language key, as in Let \( r = \) Rebecca, or Let \( r = \) the boy sitting in the right hand row of the red caboose. In such cases it will be fallacious to draw a universal conclusion (i.e. from Rebecca is a saint, we conclude everyone is a saint). Only in situations where “r” represents a randomly selected individual, may you draw a universal conclusion.
Self Practice: For each statement first determine if the Universal Introduction \( \forall I \) applies. If it does, apply the rule appropriately. Assume “r” represents a “randomly selected individual.” Odd answers p. 81

1. Ar 2. \( \sim Bc \) 3. Gr \( \vee \) Cr 4. Qr \( \equiv \) Tt 5. \( \sim (Mr \cdot Vr) \) 6. Gb \( \supset \) Nr
7. \( (\forall x)Axr \) 8. \( (\forall y) \sim Lry \) 9. \( (\forall y) (Fry \cdot \sim Sry) \) 10. \( \sim Vt \vee Ct \) 11. Rr \( \supset \) (Dr \cdot Tr) 12. \( (\exists x) Lrx \)

Existential Elimination Rule (\( \exists E \)):

Existential Elimination (\( \exists E \)) rule has two important restrictions. We use it to draw a conclusion from an existentially quantified statement, such as “Someone is happy,” or “Some will not be hired,” or “There are blue fish in my fish tank.” The common meaning behind such statements is that the “someone,” the x that has the feature, does in fact exist. (This is the force of the term “existential.)

However, from “Some students are hired,” you cannot validly infer that, “Kalani is hired,” or “The woman with the most experience is hired,” or “I am hired.” Such a conclusion is be too specific. We know the claim is true of someone, but we do not know exactly who or which one the someone or thing is. From a claim about “some x” you cannot identify or designate any specific individual – as we do with Universal Elimination. But, since you are allowed to assume there is an existing someone, you can introduce a temporary name.

We use the individual constant “t” for this purpose. Like “r,” we give “t” a special meaning. It will mean “a temporarily named individual.” You continue to reason about this individual “t” until you reach the form of the conclusion that you need. You then re-quantify the temporary individual back to “some x,” using Existential Introduction rule. We show Existential Elimination rule in the following schema:

Existential Elimination Rule (\( \exists E \)):

\[
(\exists x) \varphi x \quad \text{Restrictions: where “t” is a temporarily named individual that 1) has not been named previously within the argument context, or “t” in the conclusion; and 2) where “t” is re quantified before the proof is complete.}
\]

\[\therefore \varphi t\]

The second restriction states that you must discharge the temporary name before the proof is complete. This simply means that you must put the existential quantifier back on before the proof is complete. Failing to do so would leave the proof “open” and thus unfinished and invalid.

A Logical Equivalence Rule for Quantifiers:

Sometimes you will need to change a quantifier or move a negation sign in a quantified statement. In some cases this can be done through a sequence of steps. But in other cases you will need the Logical Equivalence rule (LE) to do this. The LE rule is a replacement rule, which means that you can replace one logically equivalent expression for one another. With the LE rule you can change an
We can recognize the equivalence of different forms of quantified expression. For example, we can hear the statement, “Not everyone will be hired,” and understand that it equates to the statement, “Someone will not be hired.” Likewise, the statement “It’s not true that some dogs read,” equates to “No dogs read.” You can learn to recognize the relationship between sentences that have the same logical meaning, often without recognizing it, but we want to make the relationships between different expressions explicit. It can be difficult to think through a logical equivalence in ordinary language. Where there is a need to disambiguate, or when we need to prove one statement has the same truth value as another, symbolization can help facilitate our reasoning. As an example, consider the meanings of the two pairs of statements:

Not everyone will be hired. and Someone will not be hired.

and

It’s not true that some dogs read. and No dogs read.

First, observe the negation in relation to the quantifier. Then observe the change of quantifier. (Remember that “No” implies both a negation and a universal quantifier.) In the first pair, “not everyone” is equivalent to “someone will not” and in the second pair, “It’s not true that some one is,” is equivalent to “no one is” or “none are.” These ordinary language equivalences reflect a logical rule about how negations and quantifiers relate to each other. This Logical Equivalence rule will be presented as pairs of logically equivalent statements that can replace one another whenever they occur.

Logical Equivalence Rule for Quantifiers (LE):

\[ \neg (\forall x) \varphi x \equiv (\exists x) \neg \varphi x \]

\[ \neg (\exists x) \varphi x \equiv (\forall x) \neg \varphi x \]

[where \( \equiv \) means “is logically equivalent to”]

The Logical Equivalence (LE) rule is presented as pair of statements, using the bi-conditional operator. The bi-conditional operator means the statements have the same truth values under the same circumstances. As a substitution rule, the LE rule allows us to substitute directly one logically equivalent statement for another. The rule can be applied to an entire statement or to any logically equivalent portion. Below are some examples.

<table>
<thead>
<tr>
<th>Original Statement</th>
<th>Logical Equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example A ( \neg (\forall x) \text{Nx} )</td>
<td>((\exists x) \neg \text{Nx})</td>
</tr>
<tr>
<td>Example B ((\forall x) \text{Nx} \lor (\exists x) \text{Gx})</td>
<td>((\forall x) \text{Nx} \lor (\forall x) \neg \text{Gx})</td>
</tr>
</tbody>
</table>

In this example we replace the entire original statement.

In this example we replace only the right-hand disjunct.
Example C illustrates that negation symbols are moved relative to the quantifier. They are never eliminated or added.

A Special Case: Standard Form Categorical Statements

You will use the LE rule in the special case of negated categorical statements. When you apply this Logical Equivalence rule to standard form categorical statements you must pay attention to meaning and keep in mind the basic form of the different types of categorical statements. For example, consider the statement:

It is not true that all politicians are liars.

This statement negates the standard universal affirmative statement: All politicians are liars.

To apply the LE rule in this special case, you need to keep in mind what exactly is being negated, and the specific formula for standard form categorical statements. The statement “It is not true that all politicians are liars” is a simple negation of the entire standard form statement “All politicians are liars.” If you let \( P_x = x \) is a politician, and \( L_x = x \) is a liar, the universal affirmative claim “All politicians are liars” would be symbolized as:

\[
(\forall x) (P_x \supset L_x)
\]

and the negated statement would be:

\[
(\forall x) (P_x \supset \sim L_x)
\]

When you apply the Logical Equivalence (LE) rule you do the same two things required of the LE rule in the case of simple quantified terms: 1) change the quantifier and 2) move the \( \sim \) symbol. But in the case of standard form categorical statements, you must do a third thing: You MUST change the logical operator to correspond with the new quantifier. In this example, the result will be the standard form existential negative statement: Some politicians are not liars.

\[
(\exists x) (P_x \cdot \sim L_x)
\]

For categorical statements the Logical Equivalence rule is important because no quantifier can be removed using \( \forall E \) or \( \exists E \) if a negation symbol is in front of it. The LE rule changes the quantifier and moves the negation inside the parenthesis, so that the quantifier can then be removed. The rule allows us to see when specific statements in English have the same logical meaning and when they do not.

In the following exercise, you will get practice working with this rule.

Exercise: Apply the Logical Equivalence rule, then translate the resulting statement into English using this language key. Study the patterns in the first examples that are done for you.

<table>
<thead>
<tr>
<th>Exercise</th>
<th>Logical Equivalence</th>
<th>Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ((\forall x) \sim A_x)</td>
<td>(\sim (\exists x) A_x)</td>
<td>(\sim A_x) or (\sim (\forall x) A_x)</td>
</tr>
<tr>
<td>2. (\sim (\exists x) B_x)</td>
<td>((\forall x) \sim B_x)</td>
<td>(\sim (\forall x) B_x)</td>
</tr>
<tr>
<td>3. (\sim (\exists x) A_x)</td>
<td>((\forall x) \sim A_x) or (\sim (\forall x) A_x)</td>
<td>Note: Move ONLY one negation. It does not matter which one you move.</td>
</tr>
</tbody>
</table>
Answers to odds on p. 81

4. \( \neg (\forall x) \neg Bx \) _________________________

5. \( (\exists x) \neg \neg Bx \) _________________________

6. \( \neg (\exists x) \neg Bx \) _________________________

7. \( \neg \neg (\forall x) Bx \) _________________________

8. \( \neg (\exists x) \neg A x \) _________________________

**Special Case:** Remember to change the logical operator to coordinate with the new logical operator

9. \( \neg (\exists x)(Bx \cdot Ax) \) _________________________

10. \( \neg (\forall x) (Bx \supset Ax) \) _________________________

11. \( (\exists x) (Bx \cdot \neg Ax) \) _________________________

12. \( \neg \neg (\exists x)(Ax \cdot Bx) \) _________________________

13. \( \neg (\forall x) (Ax \supset \neg Bx) \) _________________________

**Challenge:** The challenge here is in your powers of observation. How do these differ from 9-13 above? Do you need to change the logical operator in these problems?

14. \( \neg (\forall x) Ax \supset (\forall x) Bx \) _________________________ Use LE only on the part where it applies.

15. \( (\exists x) \neg Bx \supset (\forall x) Ax \) _________________________ Use LE on either part.

\[ \text{OR} \] _________________________

16. \( \neg (\forall x) Ax \vee (\forall x) \neg Bx \) _________________________ OR _________________________

17. \( (\exists x)Bx \cdot (\forall x) Ax \) _________________________ OR _________________________
OPERATOR RULES

A second class of rules shows how valid conclusions can be derived from statements based on their main logical operator. As with quantifier rules, these logical operator rules are either (I)nroduction rules or (E)limination rules. The Introduction rules allow you to introduce an operator, and the Elimination rules allow you to eliminate a statement from the relationship. (The vE rule is an exception to this way of thinking.) For example, Conjunction Introduction (• I) rule lets you introduce two statements in a conjunction relationship, and Conditional Introduction (=I) will let you bring two statements into a conditional relationship. Elimination rules, on the other hand, will allow you to derive part of a statement that has a given operator. So, Bi-conditional Elimination(=E) rule will specify when and how you can draw out of a bi-conditional statement from a bi-conditional relationship, and Conditional Elimination (⇒E) rule will specify when and how you can draw part of a conditional statement out of a conditional relationship.

These operator rules are shown to be valid by the truth table method. This is because the meaning of each of each logical operators is defined by a unique truth table that shows how that operator mediates truth value. These same truth tables also show for each operator when it was not possible for true premises to lead to a false conclusion. For all the operator rules, the truth table for that argument pattern, will show no rows where all premises were true and a false conclusion. Therefore, you can use these rules with the certainty that IF the premises were all true, the conclusion will follow as true.

Conjunctions Rules

A conjunction is true only when both conjuncts are true. If it contains one false conjunct, the entire conjunction statement is false.

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p • q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

Consider the argument: p, q ⊃ p • q
where p and q are premises and p • q is a conclusion drawn from them.

If you look at the truth table where p and q are both true, p • q is also true; and whenever p • q is false, at least one of the premises is false. There is no row in which BOTH of the conjuncts are true and the conclusion false. This table validates the rule called Conjunction Introduction (• I) but the rule also corresponds with a basic logical intuition, and it may seem to you that this is too trivial to count as a conclusion. Certainly, if you know two separate claims to be true, say, “Kobe is a basketball player” and...
“Sharon is a Philosophy professor,” you know that “Kobe is a basketball player and Sharon is a Philosophy professor.” This may not “feel” like a significant conclusion, but deductive reasoning often moves through a sequence of simple, sometimes trivial steps to arrive at surprising conclusions. The truth table justifies the validity of this simple logical intuition, which is formalized below.

<table>
<thead>
<tr>
<th>Conjunction Introduction Rule (• I):</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
</tr>
<tr>
<td>q</td>
</tr>
<tr>
<td>[\vdash p \cdot q] OR [\vdash q \cdot p]</td>
</tr>
</tbody>
</table>

Alternative conclusions illustrate that the conjuncts can be placed on either side of the conjunction sign.

**Conjunction Elimination Rule**

One operator rule is Conjunction Elimination (• E). Conjunction Elimination is an immediate inference, which means you can draw a direct conclusion from any statement whose main logical operator is a conjunction. Using Conjunction Elimination lets you break up any conjunction statement and conclude that both the conjuncts are true. If you think about the logic of the word, “and,” it should appear relatively obvious that if you know “Hawai‘i is a state and it is in the Pacific Ocean,” then you know (by deduction) that “Hawai‘i is a state” AND you also know, “Hawai‘i is in the Pacific Ocean.” Both these conclusions can be drawn forward or derived from the original statement. And this is all that is expressed by the Conjunction Elimination rule. The rule pattern is shown below.

<table>
<thead>
<tr>
<th>Conjunction Elimination Rule (• E):</th>
</tr>
</thead>
<tbody>
<tr>
<td>p \cdot q</td>
</tr>
<tr>
<td>p \cdot q</td>
</tr>
<tr>
<td>[\vdash p \quad \cdot E]</td>
</tr>
<tr>
<td>[\vdash q \quad \cdot E]</td>
</tr>
</tbody>
</table>

The two alternative conclusions illustrate that you can eliminate from either side of the conjunction sign.

**Using conjunction rules**

Conjunction rules are simple to use, but sequencing is important. As with any operator rule, you can apply the rule ONLY to the main logical operator. This will be the operator that is outside all punctuation. So, given the statement, \((A \lor B) \cdot (C \land D)\), you can use (•E) rule ONLY on the conjunction that links \((A \lor B)\) with \((C \land D)\). You cannot go inside the parentheses to use (•E) to derive either C or D. These statements can be derived, but only after \(C \land D\) has been derived. You cannot use conjunction Elimination on a statement like \((A \land B) \lor (C \land D)\), or \((A \land B) \supset C\) because their Main Logical Operator is not a conjunction.

35
DEDUCTIVE PROOFS: THE TWO-COLUMN METHOD

Deductive proofs illustrate the high standard required of deductive reasoning. Given the demand for certainty, deductive reasoning requires that we be clear, explicit and transparent, showing each step sequentially. Deductive proofs show how each conclusion is drawn forward from a given set of premises in a step by step process. The two-column proof presents a sequence of inferences that allow for independent and objective review at each step. Every move is a clear, simple, valid deduction that is justified by explicit reference to a rule, and set of statements that illustrate the pattern of that rule.

The two columns need to be clearly defined. Premises are presented in a left-hand column, each on a separate numbered line. (Remember that in the horizontal presentation, premises are separated with commas.) After the last of the given premise statements is listed and numbered, a right-hand column is presented beginning with a back slash, followed by the conclusion symbol.

1. (∃x) ¬Rx
2. (∀x) (Sx ⋀ Qx) ✓ (∃x) (Qx ⋀ ¬Rx)
derivation column justification column

Beneath the numbered premises, the left-hand column continues as the derivation column. In this column we put the conclusions drawn out of prior statements. Each step in this column, beyond the initial premise(s), will be a derivation, which is simply a conclusion that has been inferred from prior steps (either premises, assumptions, or previously justified derivations) and justified by reference to a specific valid rule. The exception to this will be when we enter an assumption – which will be discussed later.

Parallel to the derivation column is the justification column. To justify any line you enter in the proof, after the premises, use a shorthand notation. This shorthand references two things: the previous steps that state the premise(s) from which you drew a new conclusion, and a reference to the specific rule that validates the derivation. Use only one rule at a time.

Proofs are highly ordered structures that follow specific conventions. You need to pay attention to these conventions whenever you set up an argument for proof. First, create a left-hand column in which you list and number the premises. Each premise will be placed on a separate line, directly under each other to form the column. On the line of the final premise you put a backslash, followed by the three-dot conclusion symbol and then state the conclusion.

The conclusion statement is the goal of the proof. It will be the final line of the proof, BUT it cannot be used as a statement in the proof itself. To do so would be to commit the fallacy of “arguing in a circle,” or “begging the question.” You commit this fallacy when you assume the statement you are trying to prove. To avoid confusion it is best to put the conclusion at some distance to the right of the final premise.
**Beginning a Proof:**

The actual proof begins on the line after the final premise. This is where you begin to draw conclusions, using the given premises and valid rules, until you are able to justify the given conclusion statement. Each conclusion derived must be justified by specific reference to the rule used and the step(s) that show the pattern of the premises from that rule. A shorthand is used that consists of the step numbers and the rule abbreviation. After a conclusion is justified, it becomes a premise that can be used in further derivations. So, a proof is just a series of simple conclusions that moves from the premises toward the given conclusion in a step-by-step way until the final conclusion is reached.

After setting up the given premises and conclusion in the proper format, it is helpful to do some pre-proof planning. Do this by first looking over the premises and the conclusion statements, scanning the “landscape” of the argument. Try to find the elements of the conclusion in the premises and note the operators and quantifiers involved. This will help you see which rules you will need to use before you actually begin the proof. In the argument below, you can see that you will need to draw out the ~Q (from premise 2) and the ~R (from premise 1) and put them together using a conjunction operator. This conjunction is existentially quantified. So, you know you will need to use Conjunction Introduction rule and Existential Introduction rule. Before you can do that, however, you must eliminate the quantifiers on the premises. So, you need to review how these rules work, which means you need to remember any restrictions that may be applicable. Once this is done, you can use Conjunction Elimination rule to draw out the ~Q.

Strategic thinking is a good way to begin before you commit to any steps in the proof. It helps to focus and tune in on a handful of the rules that might be useful in the specific situation. In this argument, both quantifiers must be eliminated, and the restriction on Existential Elimination rule dictates that you should start the proof by using Existential Elimination rule on step 1. Then you can eliminate the universal quantifier on step 2. Use the same individual constant to replace the variable in both cases. Below you will see how these two steps are derived and justified in steps 3 and 4.

1. \((\exists x) \sim R_x\)
2. \((\forall x) (S_x \cdot Q_x) \quad /: \quad (\exists x) (Q_x \cdot \sim R_x)\)
3. \(\sim R_t \quad \exists E \quad 1\)
4. \(S_t \cdot Q_t \quad \forall E \quad 2\)

**REMEMBER:** Whenever you eliminate a quantifier, you **must** replace the variable. HERE, you replace the variable with “t” because WHENEVER you apply Existential Elimination rule you are, for the sake of argument, assuming a “temporary individual constant.” Because you want to create a consistent context for further argument, use “t” again to replace the variable at step 4 when you apply \(\forall E\) to step 2. The validity of this derivation depends on the meaning of “all” or “every,” which is carried by the symbol \((\forall x)\). If “Everything is both S and Q”– as is claimed by statement 2 – then it must be true that any individual you name (temporary or otherwise) is S and Q.

After quantifiers are removed, you can draw out the terms you need in the conclusion and then
combine them. You use the •E and the •I rules in the next two steps. This is shown below:

1. (\exists x) \neg Rx
2. (\forall x) (Sx \cdot Qx) \quad /.: (\exists x) (Qx \cdot \neg Rx)
3. \neg R t \quad \exists E \ 1
4. St \cdot Qt \quad \forall E \ 2
5. Qt \quad \cdot E \ 4
6. Qt \cdot \neg Rt \quad \cdot I \ 5, 3
7. (\exists x) (Qx \cdot \neg Rx) \quad \exists I \ 6

Now that you have derived a conjunction that has the same form as the conclusion, you can apply \exists I rule. This will satisfy the restriction on \exists E, which allows you to only use a a “temporary individual” to further your reasoning. That temporary individual must be re-generalized back to “someone.” The proof is completed at line 7, when you have matched the given conclusion.

Exercise: Present a simple 2 column proof that shows how the given conclusion can be derived from the given premises. (Answers to odds on p. 82-)

<table>
<thead>
<tr>
<th>Exercise:</th>
<th>Present a simple 2 column proof that shows how the given conclusion can be derived from the given premises. (Answers to odds on p. 82-)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Simple proofs with rules that have no restrictions. Each proof can be completed in 2-7 steps:</td>
<td></td>
</tr>
<tr>
<td>1. (\forall x) Nx \quad /.: (\exists x) Nx</td>
<td>You can ALWAYS derive an existentially quantified statement, from a universally quantified statement, but you must go through an individual constant.</td>
</tr>
<tr>
<td>2. (\forall x) (Nx \cdot Tx) \quad /.: (\exists x) (Nx \cdot Tx)</td>
<td></td>
</tr>
<tr>
<td>3. \neg (\exists x) Sx \cdot (\forall x) Mx \quad /.: (\exists x) (Mx \cdot \neg Sx)</td>
<td></td>
</tr>
<tr>
<td>4. (\forall x) (Nx \cdot Tx) \quad (\forall x) (Mx \cdot Sx) \quad /.: (\exists x)Nx \cdot (\exists x) Sx</td>
<td></td>
</tr>
<tr>
<td>5. (\forall x) (Nx \cdot Tx) \quad (\forall x) (Mx \cdot Sx) \quad /.: (\exists x)(Nx \cdot Sx)</td>
<td></td>
</tr>
</tbody>
</table>

Consider what other conclusions are possible to draw from the premise set of 1 & 2.
B. Quantifier rules with restrictions. 6, 7 & 14 require you pay attention to the restrictions on $\exists E$. 8&9 require you use LE rule first then pay attention to $\forall I$. 10-13 require $\forall I$.

6. $\forall x \, N_x \cdot (\forall x \, T_x)$
   $(\exists x) \, S_x$  
   $\therefore (\exists x) (N_x \cdot S_x)$

7. $\forall x \, (N_x \cdot T_x)$
   $(\exists x) \, (S_x \cdot \sim L_x)$  
   $\therefore (\exists x) (T_x \cdot \sim L_x)$

8. $\sim (\exists x) \, S_x \cdot (\forall x) \, M_x$  
   $\therefore (\forall x) \, (M_x \cdot \sim S_x)$

9. $\sim (\exists x) \, N_x$
   $\sim (\exists x) \, S_x$  
   $\therefore (\forall x) (\sim N_x \cdot \sim S_x)$

10. $(\forall x) \, (N_x \cdot T_x)$
    $(\forall x) \, (M_x \cdot S_x)$  
    $\therefore (\forall x) (N_x \cdot S_x)$

11. $(\forall x) \, N_x \cdot (\forall x) \, T_x$  
    $\therefore (\forall x) (N_x \cdot T_x)$  
    A universal quantifier can distribute in both directions.

12. $(\forall x) \, (N_x \cdot T_x)$  
    $\therefore (\forall x) \, N_x \cdot (\forall x) \, T_x$

13. $(\exists x) \, (C_x \cdot \sim V_x)$  
    $\therefore (\exists x) \, C_x \cdot (\exists x) \, \sim V_x$  
    An existential quantifier cannot distribute in both directions.

14. $(\forall x) \, (N_x \cdot T_x)$
    $(\exists x) \, (C_x \cdot \sim V_x)$  
    $\therefore (\exists x) \, (N_x \cdot \sim V_x)$

3 CHALLENGE PROBLEMS

15. $(\forall x) \, (N_x \cdot T_x)$
    $(\exists x) \, (C_x \cdot \sim V_x)$  
    $\therefore (\forall x) \, N_x \cdot (\exists x) \, C_x$

16. $\sim (\forall x) \, B_x \cdot (\exists x) \, \sim T_x$  
    $\therefore (\exists x) \, \sim B_x \cdot \sim (\forall x) \, T_x$

17. $\sim (\forall x) \, (N_x \Rightarrow T_x)$
    $(\exists x) \, \sim F_x$  
    $\therefore \sim (\forall x) \, F_x \cdot (\exists x) \, \sim T_x$

After completing these proofs, you should understand 1) how to set up a basic two-column proof, and 2) how to use all five quantifier rules – being mindful of restrictions. We will continue to introduce a system of basic inference rules. In this process you will learn rules for the conditional, bi-conditional, disjunction and negation operators. As with the conjunction rules, the validity of each rule is grounded in the definition of the logical operator and is validated by the truth table that defines that operator.
Conditional Rules

Conditional Elimination rule is perhaps the most important rule in logic. It is often known by the name, Modus Ponens, and its pattern is shown below.

**Conditional Elimination (\(\rightarrow E\)):**

\[
\begin{align*}
    p & \rightarrow q \\
    p & \\
    \therefore q & \rightarrow E
\end{align*}
\]

The validity of the rule follows from the definition of the conditional operator given in the truth table below.

<table>
<thead>
<tr>
<th>P2 p</th>
<th>(\therefore q)</th>
<th>P1 (p \rightarrow q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

The table shows that from the possible values given for a conditional statement alone (found in column three), it is not possible to draw a definitive conclusion about the truth value of \(p\) or of \(q\), because where \(p \rightarrow q\) is a true statement, the values of \(p\) and \(q\) differ. In row 1 they are both true, in row 4 they are both false, in row 3 \(p\) is false but \(q\) is true. Therefore, **no valid inference can be drawn immediately, or directly from a given conditional statement.**

Before drawing a conclusion from any given conditional statement, it is necessary to have the support of a second premise. Again, the truth table that defines the meaning of this operator shows this clearly. Only row 1 shows a way to a definitive conclusion. If any conditional statement, \(p \rightarrow q\), is presented as true AND somewhere in the same argument, its antecedent, \(p\), is presented as true, then you can conclude with absolute certainty that \(q\) (the consequent) MUST be true. This logical fact is the rule Conditional Elimination.

Conditional Elimination rule may be more easily understood by using an ordinary language example to illustrate it. Let’s consider the conditional statement, *If the Lakers finish first in the West, then they will play either Dallas or Golden State in the first round of the playoffs.*

From this conditional statement alone you know nothing about the truth value of the antecedent statement (*The Lakers finish first in the West*) or the consequent statement (*The Lakers will play wither Dallas or Golden State in the first round of the playoffs*). However, if you do know that the antecedent statement is true, then the relationship between the two statements creates a strong deductive context.
sufficient for us to derive the consequent statement as a definitive conclusion. This is easier to see when the statements are placed next to each other in a way that maps easily against the pattern of the $\Rightarrow E$ rule.

Premise 1: If the Lakers finish first in the West, then they will play either Dallas or Denver in the first round of the playoffs.
Premise 2: The Lakers finish first in the West
Conclusion: [The Lakers] will play either Dallas or Denver in the first round of the playoffs.

The argument above is a substitution instance of the pattern of inference that matches $\Rightarrow E$ rule. You can see that the second premise maps exactly upon the antecedent statement in the first premise. The conclusion statement maps exactly onto the consequent statement of the first premise. This conclusion is valid because the premises have the formal relationship illustrated by the rule. This is more easily seen when you symbolize the simple argument above. Use the following language key:

Let $A = \text{The Lakers finish first in the West}$.
$B = [\text{The Lakers}] \text{ will play either Dallas or Denver in the first round of the playoffs.}$

Symbolize the argument and show its corresponding form.

<table>
<thead>
<tr>
<th>Argument</th>
<th>Argument form</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A \Rightarrow B$</td>
<td>$p \Rightarrow q$</td>
</tr>
<tr>
<td>$A$</td>
<td>$p$</td>
</tr>
<tr>
<td>$\therefore B$</td>
<td>$\therefore q$</td>
</tr>
</tbody>
</table>

Any pair of statements that has this pattern will be a substitution instance of Conditional Elimination rule and, therefore, will be valid.

To draw a valid inference using this rule, the pattern between the premise statements must match exactly the pattern given by the premises represented in the rule. The valid relationship for the Conditional Elimination rule will hold between any conditional statement and its antecedent. A similar pattern is invalid, and is easily mistaken for Conditional Elimination. This pattern, called the fallacy of affirming the consequent looks like this:

$p \Rightarrow q$
$q$
$\therefore p$

We can show that this argument form is invalid as shown by using a truth table:
Row 3 shows that all premise statements are true and the conclusion is false. Because this argument pattern is invalid, any argument that is a substitution instance of this pattern will also be invalid. Given any conditional statement and its antecedent, you can validly derive its consequent. Given any conditional statement and its consequent, you can validly infer nothing.

**Now use Conditional Elimination rule in a simple proof.** Prove the validity of the following argument: \(~ (\exists x) \, (Gx \cdot Tx), \, (\forall x) \, Gx \, /\,: \, (\forall x) \sim Tx\)

The first thing you do when beginning any proof is list each premise consecutively to form a left-hand column. (The order of premises is arbitrary. It is acceptable to order them in any sequence you prefer.) Each premise is numbered and these numbers will be important in the proof as references. On the same line as the last premise, write a backslash, a conclusion symbol and then the conclusion to be derived.

1. \(~ (\exists x) \, (Gx \cdot Tx)\)
2. \((\forall x) \, Gx\) \, /\,: \, \sim Gx \, \sim Tx\)

Before making any derivations, take time to note the logical signs that appear in the statements, particularly the main logical operators. These give the statements their particular “shape” and give you clues about where to start and what rules you will need. Note where the conclusion terms or statements are found within the premises. It can be helpful to highlight them. Look for patterns between the statements that correspond with the rules. Think about what rules are necessary to use given the quantifiers and operators within the argument. Start with simple steps first (Quantifier elimination or LE rules, conjunction rules) because these often set you up for Conditional Elimination and other like rules.

1. \(~ (\exists x) \, (Gx \cdot Tx)\)
2. \((\forall x) \, Gx\) \, /\,: \, \sim (\forall x) \sim Tx\)
3. \((\forall x) \, (Gx \sim Tx)\) LE 1

In this proof, premise 1 has a negation sign in front of the quantifier. You need to deal with this first. So, the proof starts with the LE rule, applied to step 1. This accomplishes two important things. It puts step 1 into standard form, and it moves the negation sign in front of \(Tx\) (which is where you want it for the conclusion). Then, by changing the form of premise 1 into a conditional statement, it is easier to see how using Conditional Elimination rule will allow the \(\sim T\) term to be derived.
Now you can remove the quantifiers, but before using $\forall E$ rule on steps 2 and 3, you should note that the conclusion is universally quantified. This means that the final step of the proof will be $\forall I$, and this rule has an important restriction. You can introduce a universal quantifier only from a randomly selected individual, “r.” Therefore, in this case, you will choose “r” to replace the variable “x.”

1. $\neg (\exists x) (Gx \cdot Tx)$
2. $(\forall x) Gx$ $\therefore (\forall x) \neg Tx$
3. $(\forall x) (Gx \supset \neg Tx)$ LE 1
4. $Gr \supset \neg Tr$ $\forall E 3$ Steps 4 and 5 can be done in any order. The order
5. $Gr$ $\forall E 2$ here shows the $\supset E$ pattern more clearly.

After taking a few steps into a proof, it is good to pull back and look where you are. This can allow you see emerging patterns more clearly as a proof moves along. In this case, look closely at steps 4 and 5 to see that the pattern corresponds with the pattern of the $\supset E$ rule. Therefore, you can apply that rule. This will move closer us to the final conclusion.

1. $\neg (\exists x) (Gx \cdot Tx)$
2. $(\forall x) Gx$ $\therefore (\forall x) \neg Tx$
3. $(\forall x) (Gx \supset \neg Tx)$ LE 1
4. $Gr \supset \neg Tr$ $\forall E 3$
5. $Gr$ $\forall E 2$
6. $\neg Tr$ $\supset E 4, 5$

After $\neg Tr$ has been isolated, all that remains is to quantify that statement. Since you are arguing about a randomly selected individual, you can universally quantify $\neg Tr$.

1. $\neg (\exists x) (Gx \cdot Tx)$
2. $(\forall x) Gx$ $\therefore (\forall x) \neg Tx$
3. $(\forall x) (Gx \supset \neg Tx)$ LE 1
4. $Gr$ $\forall E 2$
5. $Gr \supset \neg Tr$ $\forall E 3$
6. $\neg Tr$ $\supset E 4, 5$
7. $(\forall x) \neg Tx$ $\forall I$ 6

The proof is finished. Each step can be checked for validity. By comparing the statement number and rule references in the justification column, you can see whether the statement in the derivation column fits the pattern of the rule named.

As you begin to create proofs for given arguments, you can return to these models to see how a rule must be executed. Similar patterns will recur. With practice your mind will begin to remember these patterns. You will want to look for similar, analogous situations, and apply rules as you have in those prior situations when you find them.
"It's virtually impossible to become proficient at a mental task without extensive practice...[I]f you repeat the same task again and again, it will eventually become automatic. Your brain will literally change so that you can complete the task without thinking about it."

Daniel Willingham, developmental psychologist

**Exercise:** Construct proof to prove the validity of the arguments below, using only rules introduced thus far. (Answers to odds p. 82-84)

1. \(\exists x (Lx \cdot Tx), (\forall x) Lx\) \(\vdash (\forall x) \sim Tx\)
2. \(\exists x \sim Fx, (\forall x) (Sx \cdot Qx)\) \(\vdash (\exists x) (Qx \cdot \sim Fx)\)
3. \((\forall x) (Sx \supset Kx), Ss\) \(\vdash \sim \sim Ks\)
4. \((\exists x) Ax \supset (\forall x) Bx, (\forall x) Ax\) \(\vdash (\exists x) Bx\)
5. \(A \cdot (B \sim C), A \supset (B \supset D)\) \(\vdash D \sim C\)
6. \((\forall x) (Zx \sim Bx), (\exists x) (Cx \cdot Zx)\) \(\vdash (\exists x) (Cx \cdot \sim Bx)\)
7. \(P \supset Q, P \supset G, M \cdot P\) \(\vdash (Q \cdot G) \cdot M\)
8. \(P \cdot Q, P \supset \sim Q\) \(\vdash \sim Q \cdot \sim Q\)
9. \((\forall x) Ux, (\forall x) (Ux \supset Gx)\) \(\vdash (\exists x) Gx\)
10. \(B \supset (C \cdot L), C \supset Q, (Q \cdot L) \supset G, B\) \(\vdash G\)
11. \(\sim (\exists x) (Gx \cdot Tx), (\forall x) Gx\) \(\vdash (\forall x) \sim Tx\)
12. \(Ae \sim Jc, Ae \cdot Bc\) \(\vdash \sim (\forall x) (Bx \supset Jx)\)
13. \(\sim (\forall x) Ux \supset (\exists x) \sim Cx, (\exists x) \sim Ux\) \(\vdash \sim (\forall x) Cx\)
14. \((\forall x) Ux \supset (\exists x) Cx, (\forall x) (Ux \cdot Gx)\) \(\vdash (\exists x) \sim Cx\)
15. \((P \supset B) \supset [O \supset (T \supset W)], (P \supset B) \cdot (O \cdot T)\) \(\vdash W\)
16. \((L \cdot P) \supset N, P \cdot L\) \(\vdash N \cdot L\)
17. \(M \cdot Z, (Z \cdot \sim H) \supset [J \supset (O \cdot Y)], J \cdot \sim H\) \(\vdash Y \cdot O\)
18. \((\exists x) Ax, (\forall x) (Ax \supset Bx), (\forall x) (Bx \supset Dx)\) \(\vdash (\exists x)(Dx \cdot Bx)\)
19. \((\forall x) (Ax \supset \sim Bx), (\exists x) (Ax \cdot \sim Cx)\) \(\vdash (\exists x)(\sim Bx \cdot \sim Cx)\)
20. \((\forall x) Bx \cdot (\exists x)Cx, (\exists x) Dx \cdot (\forall x) \sim Ax\) \(\vdash (\exists x)(Bx \cdot \sim Ax)\)
21. \(\sim (\exists x) (Ax \cdot Cx), (\forall x) (\sim Cx \supset \sim Bx), Aa\) \(\vdash \sim Ba\)
22. \(\sim (\exists x) (Ax \cdot Bx), (\exists x) (Ax \cdot Cx)\) \(\vdash \sim (\forall x) (Cx \supset Bx)\)
Conditional Introduction Rule (\(\Rightarrow I\)):

Conditional Introduction \(\Rightarrow I\) rule, shows how to introduce a conditional operator between two statements. It plays out the hypothetical, or conditional character of an if-then statement. When you assert an “if-then” statement \((p \Rightarrow q)\), you assert that the truth of the consequent statement \((q)\) is conditioned or dependent upon the truth of the antecedent statement \((p)\). You do not necessarily claim that these statements are true. Rather, you make a claim about their relationship. In a conditional statement, the antecedent statement is an assumption. In effect you say, “As long as I assume the antecedent statement (to be true), then the consequent statement will follow (as true).”

For logicians, the focus is on the relationship. To demonstrate that a conditional relationship holds between an antecedent and a consequent statement, we introduce the convention of an assumption. In a proof, assumptions are statements temporarily introduced for the purpose of furthering the proof. Once introduced, they function like premises or other derived. But because they are not part of the original premise set, at some point, they must be closed, and no statement derived within their scope can be used in further derivations.

Using Assumption Rules:

Assumptions add an element of complexity, so use them only when you cannot derive a conclusion any other way. While adding complexity, however, once an assumption is introduced, you can use them to draw forward derivations using other, simpler rules. In this way assumption rules are like power tools. Just as a power tool lets us do a specific job more easily, assumption rules let us move forward easily and efficiently. But, as with any power tool, you need to work with clear, focused attention.

When using assumptions, specific protocols must be followed to insure the validity of reasoning. Since assumptions are temporarily introduced, indenting at the point where you introduce the assumption will indicate that you are beginning a subproof. An arrow helps to keep track of each assumption introduced. Assumptions are not justified, because they are not derivations and do not follow from prior statements. Therefore, the justification column is blank. A subproof begins with the assumption and ends with the desired subgoal is justified. This closes the assumption.

Once an assumption is introduced, it is used like a temporary premise to draw forward further inferences. But, unlike premises – which are given (as true) at the outset of the proof, assumptions are not claimed to guarantee the truth of any conclusion derived under their scope. Therefore, each assumption must be closed. The conclusion derived – and justified using a specific assumption rule – is stated outside the scope of an assumption. Finally, no statement derived within the scope of a closed assumption can be used after it is closed.
Conditional Introduction Rule ($\rightarrow$I) is an assumption rule that allows us to conclude the validity of a conditional statement. Sometimes called a “conditional proof,” you begin by assuming the antecedent of a conditional statement you want to prove. The subproof closes when you derive the consequent of the conditional you want to prove. What you are proving is the conditional relationship between the antecedent (the assumption) and the consequent (the subgoal) of the conditional you ultimately want to prove. Schematically, the rule looks like this:

<table>
<thead>
<tr>
<th>Conditional Introduction Rule ($\rightarrow$I):</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rightarrow$ p</td>
</tr>
<tr>
<td>.</td>
</tr>
<tr>
<td>.</td>
</tr>
<tr>
<td>$\rightarrow$ q</td>
</tr>
<tr>
<td>$\rightarrow$ I</td>
</tr>
</tbody>
</table>

In this schema, the arrow indicates “$p$” is an assumption. The vertical dots indicate that some unknown number of steps will be taken to derive the subgoal “$q$.” The solid line under “$q$” indicates the assumption is being closed. In the case of Conditional Introduction the point is to prove the validity of the conditional relationship between the assumption and the subgoal, such that if the assumption is true, it will lead to the subgoal. This conditional statement is written directly beneath the subgoal, outside the scope of the assumption and justified by citing $\rightarrow$I.

Conditional Introduction is used to prove the validity of the classic hypothetical syllogism. This is sometimes called a “chain” argument. This style of reasoning is also sometimes called the transitive property of reasoning (if $a$ is $b$ and $b$ is $c$, then $a$ is $c$). Here is a simple example of how this reasoning is played out in a simple reasoning chain.

If Tom is happy, then Mary is happy. If Mary is happy, then John is happy. Therefore, If Tom is happy, John is happy.

Symbolized the argument looks like:

1. T $\rightarrow$ M
2. M $\rightarrow$ J  / $\therefore$ T $\rightarrow$ J
3. T $\rightarrow$I  We assume the antecedent of the conditional we want to prove.
4. M $\rightarrow$ E 1,3
5. J $\rightarrow$ E 2, 4  The subgoal is the consequent of the conditional we want to prove.
6. T $\rightarrow$ J $\rightarrow$I 3-5  State the conditional after the subgoal is reached and the assumption is closed.

In working with quantified arguments two situations will present themselves. Consider the differences in the following two examples:
All angels have wing. Nothing with wings is a devil. So, no angel is a devil.

Symbolized the argument looks like:

1. \((\forall x)(Ax \Rightarrow Wx)\)
2. \((\forall x)(Wx \Rightarrow ~Dx)\)  \(\vdash (\forall x)(Ax \Rightarrow ~Dx)\)

In this example, the main logical operators for each statement is the universal quantifier. Therefore, it is best to take the quantifiers off, replacing the variable with “\(r\)” in anticipation of adding back the Universal quantifier at the end. So, you set your schema up using “\(r\)” and it will look like this:

\[ \rightarrow Ar \]

\[ \rightarrow \sim Dr \quad \text{subgoal} \]
\[ Ar \rightarrow \sim Dr \quad \Rightarrow I \]

Then it is simply a matter of requantifying as the final step of the proof.

The following example presents important differences. Try to identify them.

1. \((\forall x) Ax = (\forall x) Wx\)
2. \((\forall x) Wx \Rightarrow (\forall x) ~Dx\)  \(\vdash (\forall x) Ax = (\forall x) ~Dx\)

In this example, the main logical operator in each statement is the conditional. The conditional we need to prove is \((\forall x) Ax \Rightarrow (\forall x) ~Dx\). The antecedent of this conditional is \((\forall x) Ax\) and the consequent is \((\forall x) ~Dx\). We can set up the schema directly from the conclusion statement as:

\[ \rightarrow (\forall x) Ax \]

\[ \rightarrow (\forall x) ~Dx \quad \text{subgoal} \]
\[ (\forall x) Ax \Rightarrow (\forall x) ~Dx \quad \Rightarrow I \]

Use these examples to decide what you must do to complete the following proofs correctly.

**Exercise A:** Set up a schema for the proof of each argument. Your schema should include
1) the assumptions you will need (Always assume the antecedent of the conditional you want to prove).
2) the statement you need to reach to close the assumption
3) a definite close to the assumption and the statement that you derive using assumption rule/strategy. On this line also state the specific rule you are using =I (Answers to odds p. 88-90)

1. \(N \Rightarrow M, \ M \Rightarrow T\)  \(\vdash N \Rightarrow T\)
2. \(P \Rightarrow Q, \ (P \cdot Q) \Rightarrow R\)  \(\vdash P \Rightarrow R\)
3. \((\forall x)(Qx \Rightarrow Fx), \ (\forall x)[Lx = (Wx \cdot Qx)]\)  \(\vdash (\forall x)(Lx \Rightarrow Fx)\)
4. \((\forall x)(Ax \Rightarrow Bx), \ (\forall x)(Ax \Rightarrow Cx)\)  \(\vdash (\forall x)[Ax \Rightarrow (Bx \cdot Cx)]\)
5. \((\forall x)(Ax \rightarrow (Bx \cdot Cx)), (\forall x)(Bx \rightarrow Ex)\)  
\(\therefore (\forall x) (Ax \rightarrow Ex)\)

6. \(P \rightarrow \sim Q, \sim Q \rightarrow R, R \rightarrow \sim B, \sim B \rightarrow \sim W\)  
\(\therefore P \rightarrow \sim W\)

**Exercise B:** Provide proofs for the six arguments in Exercise A, and for those below.

7. \(F \rightarrow (I \rightarrow H), I \rightarrow (H \rightarrow J), J \rightarrow (I \rightarrow K)\)  
\(\therefore (F \cdot I) \rightarrow K\)

8. \((F \cdot I) \rightarrow (H \rightarrow J), I \rightarrow (H \rightarrow K), (J \rightarrow K) \rightarrow L\)  
\(\therefore (F \cdot I) \rightarrow L\)

9. \((\forall x) Gx \rightarrow Gi, Gi \rightarrow El, El \rightarrow (\forall x) Ax\)  
\(\therefore (\forall x) Gx \rightarrow (\forall x) Ax\)

10. \((\forall x) (Lx \rightarrow Wx), (\forall x) [(Lx \cdot Wx) \rightarrow \sim Jx], (\forall x) (Fx \rightarrow Lx)\)  
\(\therefore (\forall x) (Fx \rightarrow \sim Jx)\)

**Challenging Proofs:**

11. \(A \rightarrow (E \rightarrow B), B \rightarrow [(E \cdot A) \rightarrow D]\)  
\(\therefore (A \cdot E) \rightarrow (B \cdot D)\)

12. \(A \rightarrow (B \rightarrow C), B \rightarrow [(E \cdot C) \rightarrow D]\)  
\(\therefore (A \cdot E) \rightarrow (B \cdot D)\)

13. \((\forall x)(Ax \rightarrow Bx)\)  
\(\therefore (\forall x)Ax \rightarrow (\forall x) Bx\)

14. \((\forall x) Sx \rightarrow (\forall x) (Tx \rightarrow Mx), (\exists x) Tx\)  
\(\therefore (\forall x) Sx \rightarrow (\exists x) Mx\)

13. \((\exists x)[ Px \rightarrow (Qx \rightarrow Rx)]\)  
\(\therefore (\exists x) [Qx \rightarrow (Px \rightarrow Rx)]\)

14. \((\forall x)[ Px \rightarrow (Qx \rightarrow \sim Rx)]\)  
\(\therefore (\exists x) [Qx \rightarrow (Px \rightarrow \sim Rx)]\)

**Arguments in translation:** First analyze the following arguments. Then translate them. Present an accurate and clear language key. Then, construct a proof of their validity.

1. If Holly goes, then Mary goes. If Jane goes than Jill goes. If Mary goes then Jane goes. So, it is certain that if Holly goes, Jill goes.

2. If I am to be a doctor, then I will need to study lots of science and get good grades. I know this because if I am to be a doctor then I will need to get into Med school, and to get into Med school, I will need good grades. And I also know that if I am to be a doctor, I will need to study lots of science.

3. All terrorists have known connections to Al Qaida. All those with known connections to Al Qaida come from Muslim countries or have Muslim connections. Therefore, all terrorists come from Muslim countries or have Muslim connections.

4. Only athletes are svelte. All athletes engage in vigorous exercise. If you engage in vigorous exercise you are healthy. So, anyone who is svelte must be healthy. *(Remember “only”)*

5. Alice wins provided Bob wins. Carol wins, if Alice wins. Carole wins only if Darius wins. So, if Bob wins, Darius wins. *(Check your translation for validity with a truth table.)*
BI-CONDITIONAL RULES

The Bi-conditional Introduction (= I) rule plays off the definition of the bi-conditional as a conjunction of two conditionals. This definition is presented in the following formula:

\[(p \equiv q) \equiv [(p \implies q) \cdot (q \implies p)]\]

The (= I) rule says that any time you have two conditional in an argument, where the antecedent and consequent statements mutually imply one another, you can introduce a bi-conditional relationship between those statements. Symbolically, the rule is presented as follows:

**Bi-conditional Introduction Rule (= I):**

\[
\begin{align*}
p & \implies q \\
q & \implies p \\
\therefore p & \equiv q \quad (= I)
\end{align*}
\]

To prove any bi-conditional, all you have to do is derive the two conditional statements that are implied by the bi-conditional relationship. For example, if you need to prove \(A \equiv B\), you simply need to derive \(A \implies B\) AND \(B \implies A\). This can be done either by isolating these statements using simple derivation rules OR by proving these conditionals using the Conditional Introduction (= I) Rule. Which of these strategies you will use will depend on the specific argument context. Let’s consider two examples.

**Example 1:**

\[
\begin{align*}
C & \implies (B \implies A) \\
(A \implies B) & \equiv C \\
C & /\vdash A \equiv B
\end{align*}
\]

As a first step, it is recommended that you analyze the conclusion to find the two conditional statements you will need to prove. Do this on the side, or on a piece of scratch paper. The two component conditionals, implied by the bi-conditional \(A \equiv B\) are \(A \implies B\) and \(B \implies A\).

The next step is to look in the premises to see if one or both of these conditional statements can be located in the premise statements. With a quick look at the premises you can find them in premises 1 and 2.

1. \(C \implies (B \implies A)\)
2. \((A \implies B) \equiv C\)
3. \(C /\vdash A \equiv B\)

In this case it is simple enough to derive \(A \implies B\) and \(B \implies A\) by using “C” with premise 2 to derive \(A \implies B\) by Bi-conditional Elimination, and with premise 1 to derive \(B \implies A\) by Conditional Elimination. The conclusion can be reached in three simple derivations.
Example 2:
C ⊨ (A ⊨ B)
(B ∧ C) ≡ A
C ∨ D  /: A = B

In this example you can see that one of the bi-conditionals A = B in premise 1 and can easily be derived if C can be derived first. However, the second bi-conditional is not present in the premises. Therefore, you will need to prove this second bi-conditional, using Conditional Introduction rule.

1. C ⊨ (A ⊨ B)
2. (B ∧ C) ≡ A
3. C ∨ D  /: A = B
4. C
5. A = B  ≡E 1, 4
6. B  Scratch paper
7. B ∧ C  ≡I 1, 4
8. A  ≡I 2,7
9. B ⊨ A  ≡I 6-8
10. A = B  ≡I 5, 9

Example 3:
A ⊨ (B ⊨ D)
(A ∧ C) ≡ B
C ∨ D  /: A = B

In this example you should look first at the conclusion and identify the two conditional statements, A ⊨ B and B ⊨ A implied by A = B. However, when you look in the premises, you cannot find these two conditionals. This means you will have to derive them using the Conditional Introduction rule. Setting up schema will isolate and identify the separate subtasks required for this proof. This will keep track of the assumptions needed, and how they will be played out. Working in this way helps to us to keep focused and work in a clear, concise and step-by-step manner. It is important to know that all this pre-proof preparation is not part of the proof. It can be worked out on scratch paper or off to the side – away from the proof.

Scratch Paper:
1. A ⊨ (B ⊨ D)
2. (A ∧ C) ≡ B
3. C ∨ D  /: A = B

Task 1: Task 2:

A  B

Each “task” must be completed and the order does not matter. Thus, there are other variations this proof could take. The proof below is just one variation.
1. A ⊢ (B ≡ D)
2. (A • C) ≡ B
3. C • D
4. C
5. D
6. A
7. B ≡ D
8. A
9. A ⊢ B
10. A
11. A • C
12. A
13. B ⊢ A
14. A ≡ B

Bi-conditional Elimination Rule (≡E):

The Bi-conditional Elimination rule (≡E) works much like ≡E rule except that you can derive from either side of the bi-conditional sign. Remember with the conditional symbol you can only derive from the right side (the consequent side) and only if you already have the antecedent (the left side). Because of the meaning of the ≡ symbol, the ≡E rule allows us to derive from either side of the bi-conditional sign if, and only if the other side is already present in the proof (either as a premise, an assumption, or a previous derivation).

<table>
<thead>
<tr>
<th>Bi-conditional Elimination Rule (≡E):</th>
</tr>
</thead>
<tbody>
<tr>
<td>p ≡ q</td>
</tr>
<tr>
<td>p</td>
</tr>
<tr>
<td>⊢ q  =E</td>
</tr>
<tr>
<td>OR</td>
</tr>
<tr>
<td>⊢ p  ≡E</td>
</tr>
</tbody>
</table>

The alternatives illustrate that it does not matter which side of the bi-conditional sign you work from. Note the difference with conditional elimination.

If p is the case, if, and only if q is the case, AND p is the case, then it is clear that q is the case. And, if p is the case if, and only if q is the case, AND q is the case, then clearly p is the case. This rule is applied exactly like Conditional Elimination. Two statements need to be included in the justification column. The only difference is that you can “attack” from either side of the bi-conditional. With the conditional sign, you can only “attack” from the left.
**Proof Practice**: Note the difference between having a bi-conditional in the premise and needing to prove a bi-conditional. *(Answers to odds pp. 92-94)*

1. \( Q = P, M \supset (P \cdot C), M \cdot F \)  \( \therefore Q \)
2. \( A = B, \ C = A, \ A \)  \( \therefore B \cdot C \)
3. \( S \cdot G, G = (F \supset Q), F \)  \( \therefore Q \cdot S \)
4. \( X \supset (Y \cdot Z), \ Z \supset (X \cdot W) \)  \( \therefore X = Z \)
5. \( (A \lor C) \cdot M, (F \equiv K) \equiv (A \lor C), T \cdot F \)  \( \therefore K \cdot M \)
6. \( (\forall x) (Px \equiv Qx), (\forall x) (Qx \equiv Rx) \)  \( \therefore (\forall x) (Px \equiv Rx) \)
7. \( P \supset B, S \supset P, Q \supset C, C \equiv S, Q \)  \( \therefore B \)
8. \( (A \lor B) \supset (C \equiv D), M \supset (A \lor B), F \supset T, T \supset M, F \cdot C \)  \( \therefore D \)
9. \( (\forall x)[(Cx \supset Dx) \equiv (Zx \cdot Lx)], (\forall x)(Lx \equiv Tx),
    (\forall x)[(Dx \equiv Cx) \equiv Tx], (\forall x)(Lx \cdot Zx) \)  \( \therefore (\forall x)(Dx \equiv Cx) \)
10. \( (T \cdot M) \supset (B \supset A), T \cdot Z, T \supset (A \supset B), (A \supset B) \supset M \)  \( \therefore A \equiv B \)
11. \( (L \equiv \sim M) \cdot (L \supset T), T \supset \sim M \)  \( \therefore \sim M \equiv (L \cdot T) \)
12. \( B \supset (T \cdot A), D \supset T, A \supset B, (B \equiv A) \supset C \)  \( \therefore D \supset (C \cdot T) \)

**WORKING WITH DISJUNCTIONS**

Disjunction statements present alternatives. They claim that either one disjunct is the case or the other is the case, perhaps both. Given only the claim that a disjunction is the case, you cannot know which of its disjuncts are true. Therefore, no conclusions can directly be drawn out of a disjunction statement. Yet, for any given disjunction, you do know that at least one disjunct MUST be true. (This is indicated in the first three rows of the truth table that defines the disjunction operator.) Two important rules of inference can be developed from these facts.

**Disjunction Introduction Rule (\( \lor I \)):**

Disjunction Introduction rule (\( \lor I \)) is a simple rule to execute, but it presents problems for many students. Sometimes called the “Addition Rule,” it lets us introduce to any given statement, another statement using the “\( \lor \)” operator. In English this is like saying, “John is a student (is true),” therefore I can conclude “John is a student or the stock market crashed in 2001.” The validity of this
move is perplexing because the disjunct introduced sometimes seems to come out of nowhere. But the validity of this rule is grounded in the definition of this operator. As long as one disjunct is true, the entire disjunct is true, regardless of the truth value of the added statement.

Again, the truth table that defines the inclusive disjunction operator illustrates the point. Whether you begin with “p” or “q,” either as a premise or assumption, you will end up with a true disjunction statement. This is regardless of whether you add a true statement (as in row 1) or a false statement (as in rows 2 and 3.)

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p v q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

**Disjunction Introduction Rule (vI):**

\[
\begin{align*}
p & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \q
Disjunction Elimination Rule (vE):

Disjunction Elimination rule (vE) plays out the logic of a disjunctive statement. Any disjunction claims that at least one of the disjuncts must be true. You do not necessarily know, however, which one is true. Because you have no way of affirming either disjunct, you cannot derive either disjunct from the disjunction relationship. However, since you do know that one of them is true, you can “test” each disjunct by presenting it as an assumption to see what can be derived from them. If you are able to derive a common statement (say, “r”) from both disjuncts you can deduce that “r” must be true. You must enter each disjunct into the proof as separate assumptions and each must be worked independently to this common statement you want to prove.

To use vE rule, you must have a disjunction statement to begin with – either as a premise, a derivation or as an assumption. The following schema shows how the rule works:

<table>
<thead>
<tr>
<th>Disjunction Elimination Rule (vE):</th>
</tr>
</thead>
<tbody>
<tr>
<td>p v q</td>
</tr>
<tr>
<td>This disjunction must be given.</td>
</tr>
</tbody>
</table>

| Subproof 1:                     |
| p                                |
| This assumption comes from the first disjunct. |
|                                  |
| r                                |
| subgoal “r” represents the statement you want to prove. |

| Subproof 2                     |
| q                                |
| This assumption comes from the second disjunct. |
|                                  |
| r                                |
| subgoal (This subgoal must be the same as that of the first subproof |
| r vE                             |
| This third assertion of “r” is outside the scope of all assumptions. |

The reasoning of this rule goes like this: I am told this disjunction is true. Therefore, I know one of its disjuncts must be true. But, I do not know which one. However, if I can draw a common conclusion from each disjunct, knowing that at least one of them IS true, I can deduce that the common conclusion is true.

Disjunction Elimination requires two assumptions. Each is introduced separately and carried out to the same conclusion independently. For each assumption that common statement, “r,” closes the assumption, BUT it does NOT complete the reasoning strategy. The deduction is complete only after “r” is derived a second time. After closing the second subproof, you must then assert the common statement “r” for one final time, this time outside the scope of any assumption.
A schema provides a clear blueprint for the proof. It shows the disjunction that will be the source of assumptions. It shows each disjunct will be an assumption, introduced separately and taken to a common statement. (Generally this common statement will be the conclusion). Finally, it shows that common statement outside the scope of any assumption, justified by the rule vE. You illustrate this rule using a common deductive argument form called “constructive dilemma.”

1. A ⊃ B
2. C ⊃ D
3. A v C /: B v D

¬4. A
5. B ⊃E 1, 4
6. B v D v I 5

¬7. C
8. D ⊃E 2, 7
9. B v D v I 8
10. B v D vE 3, 4-6, 7-9

**Exercise A:** Using the vE pattern presented above, set up a schema for the arguments below, show the required assumptions and the common subgoal for each assumption. Be sure to complete each subproof before using the vE rule. *(Answers / hints p. 95-96)*

1. C ⊃ D, E ⊃ F, E v C /: D v F
2. R ⊃ T, P ⊃ T, R v P /: T
3. (E v F) ⊃ (C · D), (D v G) ⊃ H, E v G /: H
4. (∃x)(Ax v Bx), (∀x)(Bx ⊃ Cx) /: (∃x)(Ax v Cx)
5. (∃x)(Vx v Wx), (∀x)(Zx ⊃ Wx) /: (∃x)(Zx v Vx)

**Exercise B:** Use vE rule to execute a complete proof for each of the arguments above. Then continue to practice with the arguments below.

6. ~ P ⇔ B, K v ~ B, K ⊃ D /: D v ~ P
7. (∀x)(Gx = Mx), (∃x)(Gx v Mx), (∀x)[Gx ⊃ (Mx ⊃ Tx)] /: (∃x) Tx
8. ~ Q ⊃ (M · B), ~T ⊃ (B · H), ~ Q v ~ T /: B
9. (∀x)(Xx ⊃ Yx) · (∀x)(Zx ⊃ ~ Wx), (∀x)(Xx v Zx) /: (∃x) (Yx v ~Wx)

**Arguments in translation:** Translate the following arguments presenting an accurate and clear language key. Then construct a proof of their validity.
1. If teachers are furloughed on instructional days, then the quality of public education will decline. If teachers are furloughed on planning days, the quality of public education will decline. Either teachers will be furloughed on instructional days or planning days. Therefore, the quality of public education will decline.

2. Jo’s either going to college or travel the world since he will go to college if and only if his parents can afford the tuition. Jo will get a job only if he can use the money to travel the world. Either his parents can afford the tuition or he gets a job.

3. If Barack Obama is citizen, then he can produce a valid birth certificate. If he is not a citizen, then he is not the legitimate president of the United States. He is either a citizen or not. So, he should be able to produce a valid birth certificate, or he’s not the legitimate president.
WORKING WITH NEGATIONS

"Everything is either true, or not true, or both true and not true, or neither true nor not true; that is the Buddha’s teaching."

Working with negations is one of the more challenging aspects of logical reasoning. Given the assumption of a binary logic, that a statement can only be either true or false, a change of truth value is the difference between a true and a false answer. Negations, therefore, have significant logical importance and you need to pay careful attention to them. If a statement holds a negation operator throughout the argument context, there is little difficulty. However, when the value of a statement changes, (from affirmative to negative or negative to affirmative) within a given context, it is easy to lose track of what is being claimed and become confused.

Only one rule allows us to change truth value. It is an assumption rule, and depending on whether you assume an affirmative statement or a negative statement, you call the rule Negation Introduction (~I) or Negation Elimination (~E). This rule is also called reductio ad absurdum (to reduce to absurdity), indirect proof, or argument from contradiction. This reasoning strategy has been used for over 2500 years, and it is a cornerstone of reasoning in the history of Western mathematics and philosophy. It argues that if a given statement leads to a contradiction (or an absurdity), then that given statement cannot possibly be true. And, since every statement must be either true or false, its opposite, or negation, must be true.

The validity of the strategy derives from a basic logical principle called the Law of non-contradiction. This law states that it is not possible for a given statement to be true AND false (in the same sense) at the same time. Formally, you present this principle as the tautology, \(~ (p \cdot \sim p)\), or: It is not true that \(p\) is both true and not true.

Earlier, you used truth tables to characterize statements according to their truth value possibilities. The term contradiction named a statement that was necessarily false by virtue of its form. In ordinary language you use the term contradiction more loosely to indicate a pair of statements that are in opposition or contrary to one another. In a reductio proof you make use of a specific contradiction having the form, \(p \cdot \sim p\). This form of statement will always be false.

Although there may be contexts in which a statement like, “I am happy and I am not happy” makes perfect sense, (because the term “happy” is used in two different ways), in formal logic these nuances are not considered. Only the truth-functional aspects of the statement are taken into account. And, if you think about it from a logical perspective, If you did allow that a statement could be both true and false at the same time (thus making \(p \cdot \sim p\) a true statement), our reasoning would go nowhere. Anything could be argued to be true and anything could be argued to be false.
Everything would be equally true and false, and reasoning and insanity would be equivalent.

If the Law of non-contradiction is a logical fact, then no valid argument with a set of true premises can lead to a false conclusion. Therefore, it is possible to introduce some statement as an assumption and test to see if, while it is operative – while it is an open assumption, a contradiction can be derived. If a contradiction can be derived then, the statement that led to the contradiction must be false. Hence, the statement that has the opposite truth value of the assumption must be true. The schema for this strategy of reasoning is as follows:

<table>
<thead>
<tr>
<th>Negation Introduction (~I) / Negation Elimination Rule (~E)*</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \rightarrow p )</td>
</tr>
<tr>
<td>.</td>
</tr>
<tr>
<td>.</td>
</tr>
<tr>
<td>( x \cdot \sim x )</td>
</tr>
<tr>
<td>( \sim p ) ( \sim I )</td>
</tr>
</tbody>
</table>

* Note the difference between ~I and ~E reflects whether you introduce or eliminate the negation at the end of the subproof. This will depend upon the assumption you introduce.

Two variations are presented to show that the basic strategy is the same: Some assumption (\( p \) or \( \sim p \)) is introduced. Derivations are taken until an explicit contradiction (of the form \( x \cdot \sim x \)) is reached. At this point the assumption is closed and its opposite (\( \sim p \) or \( p \)) is asserted. Using \( x \cdot \sim x \) indicates that any contradiction can be derived. It need not be a contradiction of the assumption and its opposite (although this is often the case), nor of any particular statement. Thus, \( x \cdot \sim x \) represents ANY contradiction whatsoever.

Let’s look at an example of this strategy in action. Many of us are familiar with a simple demonstration from beginning Chemistry courses that uses litmus paper to determine whether a given solution is an acid or not. Our reasoning in testing a given sample uses this pattern: If this solution is an acid, it will turn the litmus paper red. The solution tested did not turn the litmus paper red. So, the solution is not an acid. In Chemistry class, you take the logic of this reasoning for granted. In Logic, you want to prove the form of our reasoning to be valid.

Symbolize the argument as: \( A \supset R, \sim R \vdash \sim A \), where “A” represents “The solution is an acid,” and “R” represents “The solution turns the litmus paper red. Now, demonstrate that the conclusion follows by introducing “A” as an assumption and then deriving a contradiction. This is illustrated in the following indirect proof.
1. \( A \supset R \)
2. \( \sim R \) \quad \therefore \sim A

\[ \rightarrow \]
3. \( A \) \quad \text{Assumption for } \sim I
4. \( R \) \quad \because E \quad 1, 3
5. \( R \cdot \sim R \) \quad \cdot \quad I \quad 4, 2
6. \( \sim A \) \quad \sim I \quad 3 \cdot 5

Assuming \( A \) – the opposite of the conclusion you really want to prove – you can easily draw out the contradiction between \( R \) and \( \sim R \). But a contraction is a necessarily false statement. Arriving at a false statement, in a valid argument, means that one of the statements in the argument must be false. The premise statements are assumed to be true. Therefore, the only statement that could lead to a falsehood would be the assumption, \( A \), which was introduced. Therefore, \( A \) cannot be true. Since \( A \) cannot be true, \( \sim A \) must be true.

This \textit{indirect proof} or \textit{reductio} method is a powerful technique that will let you complete proofs that otherwise have no solution. You will find further use for this approach as you attempt more challenging proofs. The following exercise will help you practice the technique.

**Exercise:** Look for the pattern of proof above in the arguments below, as you apply the \( \sim I \) and \( \sim E \) rule to the following arguments. \((\text{Answers/hints p. 97})\)

| 1. \((\forall x)(Lx \supset Mx)\), \((\exists x)\sim Mx\) | \(\because (\exists x)\sim Lx\)
| 2. \(P \supset \sim P\) | \(\therefore \sim P\)
|\hline
| 3. \(\sim A \supset B\), \(B \supset A\) | \(\therefore A\)
| 4. \((\forall x)[(Sx \vee Tx) \supset \sim Sx]\) | \(\therefore (\forall x) \sim Sx\)
|\hline
| 5. \(B \supset D\), \(D \supset E\) | \(\therefore \sim (B \cdot \sim E)\)
| 6. \(\sim Ea \cdot Ma\), \((\forall x)(Cx \supset Ex)\) | \(\therefore (\exists x) \sim Cx\)
|\hline
| 7. \((A \vee B) \supset \sim C\), \(\sim D \supset (C \cdot A)\) | \(\therefore \sim (\sim D \cdot A)\)
| 8. \((A \supset B) \cdot (\sim B \supset C)\), \(\sim B\) | \(\therefore C \cdot \sim A\)
|\hline
| 9. \((\exists x)Tx \supset (\forall x)Cx\), \((\forall x) \sim Cx\) | \(\therefore (\exists x) \sim Tx\)
| 10. \((\forall x)(\sim Px \supset \sim Tx)\), \((\exists x)(Tx \cdot Qx)\) | \(\therefore (\exists x) (Qx \cdot Px)\)

**Arguments in translation:** Translate the following arguments presenting an accurate and clear language key. Then construct a proof of their validity.

1. The Rainbow Warriors will play in a bowl game if, and only if they have a winning season. It’s obvious they did not have a winning season, because they didn’t play a bowl game.
2. If the child is autistic, then the child will not spontaneously express empathy. The child does spontaneously express empathy. We can conclude, the child is not autistic.

3. No one who has been educated about the truth of drug use will use drugs. Many who have gone through the DARE program do in fact go on to use drugs. Therefore, it seems conclusive, that some who have gone through the DARE program are not truthfully educated about drug use.

**Application Challenge: Proving Double Negation**

In formal logic, theorems are statements that are proven without given premises. Rather, it is proven from a hypothesis. The theorems you will work with are all tautologies that are expressed as bi-conditional statements. They will be proven valid, not from a set of premises, as you have done with arguments, but from assumptions alone.

In this exercise, the *reductio* technique is used to prove a basic theorem of deductive logic, Double Negation. The principle of double negation is the same in Logic as it is in mathematics and in grammar -- namely that when two negations are applied to the same statement, one negation cancels the other out. A double negated claim, therefore, holds the truth value of the original claim, as you show in the truth table below.

<table>
<thead>
<tr>
<th></th>
<th>~p</th>
<th>~ ~ p</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
</tbody>
</table>

In Logic the principle of double negation is presented in the logical equivalence, \( p \equiv \sim \sim p \). This can be read as “\( p \) is true if, and only if it is not true that \( p \) is not true.”

You use a basic proof strategy to validate the logical equivalence, \( p \equiv \sim \sim p \). This strategy will be used to develop other logically equivalent relationships that will function as replacement rules. In this strategy there are no premises to draw upon to begin deriving inferences. Therefore, you must begin the proof with an assumption. To know what you must assume as the starting point of our proof, you begin with a careful analysis that pulls out the full meaning of the bi-conditional operator. This process of analysis will give us the assumptions you need to begin the proof.

Any bi-conditional statement is an abbreviation for a conjunction of two conditionals, in which the antecedent of one conditional is the consequent of the other and the consequent of the one is the antecedent of the other. So, if you can identify the two component conditional statements implied by the bi-conditional, \( p \equiv \sim \sim p \), you can present the bi-conditional as a valid inference derived from assumptions only. You begin by analyzing \( p \equiv \sim \sim p \) into its component conditionals.
Analysis: \( p \equiv \sim \sim p \) is an abbreviation for \((p \rightarrow \sim \sim p) \cdot (\sim \sim p \rightarrow p)\)

Therefore, using \( \rightarrow I \)

\[
\begin{array}{ccc}
\rightarrow p & \text{AND} & \rightarrow \sim \sim p \\
\sim \sim p & p \\
\end{array}
\]

Therefore, using \( \equiv I \)

\[
p \equiv \sim \sim p \quad \equiv E
\]

You get our assumptions from the conditional statements that are implicit in the bi-conditional. Each assumption must be worked out independently. Once the two conditional statements have been proven, it is a simple step to assert the bi-conditional relationship. The trick will be to figure out how to get from \( p \) to \( \sim \sim p \), and from \( \sim \sim p \) to \( p \).

**Bonus Bank:** Using the schema above, and prove that \( p \equiv \sim \sim p \) is a valid theorem. Hint: Step 1 of your proof will be an assumption, which one is your choice. The first step of your proof will be one of the assumptions above. A secondary assumption will need to be introduced immediately. THINK—what assumption will work?

**A Challenging Application: Proving Disjunctive Syllogism**

The Disjunctive Syllogism is obviously valid to many of us, but it is one of the trickiest to prove valid. The premises of this simple argument form state that some disjunction is true and that one of the disjuncts is false. You use this pattern of reasoning in the following argument:

I must take either log or math. I will not take math. So, I will take logic.

The informal (ordinary language) proof follows a close consideration of the logical operators.

1. It is true that I must take either logic or math. (premise)
2. It is true that I won't take math. (premise)
3. Let's assume I take logic. (Assuming this is true, it is true as long as I reason under the scope of its assumption. Since it is the desired conclusion, I can close this assumption.)
4. Let's assume that I take math. (At this point the assumptions at 4 and 5 are open.)
5. AND Let's also assume that I don't take logic. (At this point the assumptions at 4 and 5 are open.)
6. But from what has been given as true (line 2), and assumed true (line 4), it follows that I both take and don't take math. But this is absurd.
7. Therefore, it must be false that I don't take logic (line 5).
8. Therefore, it must be true that I do take logic. (By the theorem Double Negation)
9. Therefore, it must be true that I take logic.
Here is the formal proof:

1. \( L \lor M \)
2. \( \sim M \)
   \(/: L\)
3. \( L \) Assumption and close1 for \( \lor \)E
4. \( M \) Assumption for \( \lor \)E
5. \( \sim L \) Assumption for \( \sim I \)
6. \( M \cdot \sim M \)
   \(/: L \)
7. \( \sim \sim L \)
6, 4 \ sim I 2, 4
8. \( L \)
   \(/: L \)
7. \( \sim \sim L \)
5-6 close \( \sim I \)
9. \( L \)
   \(/: L \)
8. DN 7
close 2 for \( \lor \)E
10. \( L \lor E \)
7, 8 final close \( \lor \)E

This proof is used as a subproof in some more challenging proofs in the exercises below. It can be used whenever you need to demonstrate the truth of one disjunct when you know the other disjunct is false.

DEVELOPING PROOFS: SKILLS AND STRATEGIES

You now have a set of rules governing both quantifiers and logical operators. This set of rules are sufficient to prove any valid deductive argument presented in our logical languages. In this sense you have developed a complete deductive system. You have also shown how the basic operator rules can prove a range of theorems that can, in turn, function as Replacement rules. Replacement rules let us replace one expression with another logically equivalent expression. They can be applied to an entire statement, or to any part of a statement. Adding theorems to our rule set allows us to expand the possibilities for making inferences in ways that open up the possibility to construct proofs more efficiently and elegantly.

You have learned a basic two-column proof method. This method is a powerful tool, but to use it at a high level is a skill requiring knowledge of the conventions established for presenting proofs, and a combination of aptitude and personal practice.

As you practice you become familiar with recurring patterns and sequences of moves, and you can recognize more readily when a given rule or set of rules applies. The more you practice, the more you will be able to anticipate how sequences of steps play out. To use an analogy, it is like finding your way around in a strange city. At first everything looks strange and you get lost. You make wrong turns and end up in the wrong place, sometimes going far out of the way you intended. But the more familiar you become with the streets and the landmarks, with the way the city is organized, the less you have to consciously think about how to get from one place to another. Eventually you become so
comfortable, so familiar with the layout of the city, you just go where you want to go without having to think through how you will get there.

Working with proofs is similar. There is often a steep learning curve. Practice can be frustrating. But with persistence you recognize repeating patterns and relationships. Your mind begins to anticipate steps that follow from these patterns. In this way the skill of constructing proofs is much like playing any deductive game. Practice gives us the kind of familiarity our minds need to recognize the basic patterns that are the rules themselves. Eventually you begin to see these basic patterns embedded within increasingly more complex contexts.

There will always be situations where the way forward is not readily seen. In such situations, sketching out a plan of action, using a schema, narrowing the options that are available given the operators in play, are ways to in which you can think through a proof before beginning to commit it to paper. One important tactic to keep in mind, is to leave a problem alone when you feel “stuck.” Going onto another problem, or leaving the effort entirely by walking away and doing something else for a while (something relaxing) can help clear our minds. Often when you return, you will find you can more easily see a solution. Our minds seem to be hardwired for deductive reasoning and our brains are able to continue working on problem solving without our conscious effort.

Understanding some basic strategies can help to think through a given situation at the outset. Everyone develops their own approach to proofs. Not all the strategies discussed below will appeal to everyone equally. They are not the only ones that can be used. Deductive thinking can benefit from a methodical approach but it also requires creativity. So, use them as you find they help you in your practice and be adventurous enough to discover your own.

**Scratching and Sketching**

Except in the simplest problems, you should not expect to “see” the entire proof at the outset. With the exception of proofs that require just two or three steps, it is worthwhile sketch out on scratch paper some of the possibilities you see in the patterns of statements within a given argument. Far too often students believe their first attempt must lead to a valid solution. Not So! Being flexible and willing to drop an initial strategy when you see it may not be working is often necessary, particularly when problems involve many steps, or multiple assumptions. Experienced “proofers” make false starts, find that an initial “hunch” isn’t panning out, misread and make mistakes. However, it is OFTEN in the course of these errors that a valid solution becomes clear. So, don’t be afraid to make mistakes!

Taking time at the outset to look over the problem, to see which operators are involved, where the negation signs are (do they move or stay attached to a given statement), which statements repeat,
and which are drawn out of the premises and into the conclusion– these sorts of observations can be made before pencil is put to paper.

**Do What You Can Do Now**

The “*Do What You Can Do Now*” strategy involves drawing whatever inferences are available as you recognize a pattern that corresponds to one of the rules that you have learned. It goes without saying that to use this basic strategy, you must first be able to recognize the rule patterns so that they get into your brain and you can recognize them when they are present. A good way to begin the “*Do what you can do now*” strategy is to look over the premises and see if you have conjunctions. Since •E rule is an immediate inference that is simple to apply, finding premises that are conjunctions, allows you to derive two conclusions in two successive steps. This often gives you statements that can be used with others in the execution of ⇒E or ≡E.

Quantifier rules are all immediate inferences so, if the premises are quantified, you can always begin by eliminating quantifiers or adding them to set up for further rules, if this is necessary. Always look to see if any pair of statements or parts of statements match other statements in an argument in a way that reflects a ⇒E or ≡E pattern. If a pair matches a given rule, then simply begin by using that rule to derive a conclusion.

“If you derived it, can you use it,” is a variation of this strategy. Often when you derive a statement you can immediately use it with another statement to derive a further conclusion.

**Divide and Conquer**

“*Divide and Conquer*” is a basic problem solving strategy that applies to a wide range of problems (in and out of logic). When confronted with any complex problem, it is often helpful to break it into smaller problems that can be solved separately. This strategy is particularly useful when dealing with conclusions that are conjunctions or disjunctions. Whether conjunctions are presented as premises or as a conclusion, the conjunction rules allow us to easily draw out needed statements, and put any two available statements together in a conjunction. So, if there are premises that are conjunctions (as their main logical operator), it is a good rule-of-thumb to start by breaking them up using the •E rule. This move increases the number of statements you have to work with, and brings forward new relationships among the available statements. From these relationships you can often see patterns that validate further inferences.

When the conclusion is a conjunction statement (and this conjunction cannot be directly derived from the premises), try to break up the conclusion into its separate conjuncts and solve for each conjunct independently. Then, the final step is simply a matter of applying the •I rule. Somewhat
similarly, if the conclusion is a disjunction, you can derive just one side, one disjunct. The other can then be added on using vI.

**Working Backwards**

"Working Backwards" works best in situations when no assumptions are required. This strategy is best suited to simpler arguments. It is easy to start because the final step is never in question. It is ALWAYS the given conclusion of the argument. It is also often easy to identify the specific inference rule that will justify this step. Furthermore, since all derivations are either directly made from a single prior statement, or a pair of statements, you can almost always identify at least one of the statements from which the conclusion must be derived.

Working backwards begins by identifying the conclusion as the final step of the proof and thinking: What rule will allow me to conclude this statement? Then ask: Given the relation this statement has, what logical rule will let me derive it?

In the argument below it is easy to identify that, at final some step “n,” (where “n” is just some numbered step), ∼T will be derived. ∼T is found in premise 2 where it is in a conjunction with E. In order to derive ∼T, you must first derive (E · ∼ T). But this conjunction can only be derived from step 2 using ⊃E – because it is the consequent of a conditional statement. But before this can happen, however, you must derive C v D – since you can only derive the consequent of a conditional if you first have its antecedent. This analysis gives you three backward steps.

If you write down ∼T at the very bottom as step “n”. In the justification column you can put ⊃ E, the rule you will use to derive ∼T. On the step directly above it, as line “n-1,” you write E · ∼ T and in the justification column you can write ⊃E 2, since you know that E · ∼ T will come from step 2 and that it will come by the rule ⊃E. If you think back one step more, you know that before you can use ⊃E, you must have the antecedent of statement 2 before you can derive E · ∼ T. So, you can write C v D, directly above E · ∼ T as step “n-2.”

1. (A · B) ⊃ C
2. (C v D) ⊃ (E · ∼ T)
3. B · A          / ∴ ∼ T

While you can deduce these backward steps relatively easily, you don’t yet know which step
(numbers) will complete the justification for these moves. In the case of $C \lor D$, you may not even be able to figure out what rule will allow us to derive it, even if you know it is necessary that it be in play before $E \cdot \sim T$ can be derived from premise 2 using $\Rightarrow E$. This should not be a concern. It is not always necessary to know exactly how you will derive all the statements you need. What the sketch shows are the final steps that must necessarily be taken to finish the proof.

At this point you may not feel you can go further with the “Backward Strategy.” No worry. You can go back to the premises and note that you have not used premises 1 and 3. Premise 3 is a conjunction, $A \cdot B$ and $B \cdot A$ is in premise 1. Conjunctions can easily be broken up and rearranged using the conjunction rules so you can now play with the “Do What You Can Do Now” strategy. If you work in this way— and keep thinking— the path to the proof often becomes clear. Generally, between these two tactics – working from the conclusion backwards and working from the premises down, all the steps required to derive the conclusion validly can be identified in the proper sequence. All that remains to complete the proof is to fill in the justification column appropriately.

**Working with Assumptions: Using Schema**

Working with assumptions always adds an element of complexity to a proof. When you introduce an assumption, several things must be kept in mind. First, while you can introduce assumptions into a proof at any time, and you can introduce as many assumptions into a proof as you want (and hopefully need), you are still bound by a rule defined context and you must close every assumption in accord with one of the assumption rules ($\Rightarrow I$, $\lor E$, and $\sim I$ or $\sim E$). You must also remember that after an assumption is closed, no statement derived under its scope can be used in further deductions. This makes sequencing assumptions important.

The strategy of sequencing assumptions within the scope of another open assumption is called “nesting.” When you nest assumptions, you must close them in reverse order. With proofs that require nesting or involve multiple assumptions, a “schema” can help you navigate your way through the best way to open and close each assumption.

You can easily “schematize” your assumptions for proving conditional statements, even quite complex conditional statements, by focusing on the main logical operator, and pealing off the antecedent statement to introduce as an assumption. The consequent will be the subgoal. If that subgoal is also a conditional statement, you simply repeat the process inside the open assumption.

As an example, say you want to prove a complicated conditional: $A \Rightarrow \{B \Rightarrow [C \Rightarrow (D \Rightarrow E)]\}$. Because the $\Rightarrow I$ rule requires you assume the antecedent of the conditional you are trying to prove and try to derive the consequent, you can set up the schema by putting the statement you want to prove at the bottom. Draw a line above it. Then, just peel off the antecedent (in this case “A”) and place it
above, indicating it is an assumption with an arrow (→). Then place the consequent of that antecedent above the subgoal line. Now, look to the subgoal and repeat the process, until you have introduced an assumption that will allow you to begin deriving new statements.

This “nesting” process can be used quite effectively. The key to figuring out what a secondary or tertiary assumption must be, is to look to the subgoal and ask yourself what will allow you to draw that subgoal forward. Sketching out your strategy using schema will help you see clearly what is being assumed and what needs to be derived (the sub (or subsub) goal) before that assumption can be closed. If you are not sure what to present as a next assumption (but think you might need one) you can look to the subgoal to get a clue.

The strategy of assumptions can be schematized as follows:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
<td>C</td>
<td>D</td>
<td>E</td>
</tr>
</tbody>
</table>

The difficult part of nesting is figuring out the order for introducing each assumption. This difficulty can be dealt with by being clear about the statement you want to ultimately derive, what rule must be used to derive such a statement, and what the necessary assumption and subgoal are required by that rule. If your first assumption strategy does not seem to be working out, a good strategy is to change the order of assumptions. Often this simple adjustment allows the proof to play out.

Parallel Assumptions:

Other assumption strategies work in a parallel way. This strategy works well when you know that you must make two separate assumptions. Each assumption must be played out completely before the second is introduced. You use this approach with the Disjunction Elimination rule, and also at times in proving bi-conditional statements when both conditionals need to be proven through assumption. In the following proof, we combine divide and conquer strategy with a parallel assumption strategy.

\[ \neg C \cdot \neg D \]
\[ (A \rightarrow C) \cdot (B \rightarrow D) \quad \therefore \neg A \cdot \neg B \]
**Strategy:** derive ~ A and ~B separately; join them at the end

**Schema:**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>~A</td>
<td>~I</td>
<td>~B</td>
</tr>
</tbody>
</table>

**Final conclusion:** ~ A · ~ B · I

In “parallel strategy” each part of the conclusion is derived independently, through assumption. The way each part is derived similar, but the first assumption must be closed before opening the other.

Using schema requires you pull back and carefully consider the relationships and the details that define the problem BEFORE committing to a proof. This effort can save time. The mental effort goes into working out an effective schema. Once you have a good plan, the steps required to realize it are – more often than not – easily executed.

All rules and strategies have now been presented. The following exercises are intended to give you practice in using and extending your skills.

**BONUS BANK ASSIGNMENT**

At this point all techniques and strategies necessary for proving the Rules of Replacement have been presented. While it is not for the faint of heart, YOU MAY RECEIVE Bonus Bank for providing a proofs for the following Logical Theorems. Each proof will be worth 1 or 2 points. To receive Bonus Bank credit, you need to independently complete a proof and then discuss it with the instructor.

(All proofs follow the same basic pattern. See proof of Double Negation above for a model of how to set up and execute these proofs. All proofs will start with an assumption. The first assumption will always come from your analysis of the bi-conditional you must prove in each case.)

<table>
<thead>
<tr>
<th>Name of Theorem</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ((p \cdot q) = (q \cdot p))</td>
<td>Law of Commutation for (\cdot)</td>
</tr>
<tr>
<td>2. ((p \cdot q) \cdot r = p \cdot (q \cdot r))</td>
<td>Law of Association for (\cdot)</td>
</tr>
</tbody>
</table>
3. \((p \lor q) \equiv (q \lor p)\)  \text{Law of Commutation for } \lor \text{ v comm}

4. \((p \lor q) \land r \equiv p \lor (q \land r)\)  \text{Law of Association for } \land \text{ v assoc}

5. \((p \supset q) \equiv (\neg q \supset \neg p)\)

6. \([p \supset (q \supset r)] \equiv [(p \land q) \supset r]\)

Transposition  exportation  trans  exp

The following proofs require sustained thought and planning in the use of Negation Introduction/Elimination Rule.

7. \((p \supset q) \equiv (\neg p \lor q)\)  \text{Implication } \text{imp}

8. \([p \land (q \lor r)] \equiv (p \lor q) \land (p \lor r)\)  \text{Distribution for } \land \text{ dist } \land

9. \([p \lor (q \land r)] \equiv [(p \lor q) \land (p \lor r)]\)  \text{Distribution for } \lor \text{ dist } \lor

10. \(\neg (p \land q) \equiv (\neg p \lor \neg q)\)  \text{De Morgan for } \land \text{ deM } \land

11. \(\neg (p \lor q) \equiv (\neg p \land \neg q)\)  \text{De Morgan for } \lor \text{ deM } \lor

Developing and Expanding Proof Skills: Practice Exercises

Exercise A: Prove the validity of the following arguments.

1. \((\forall x) (F x \supset \neg G x), \ Ga \quad /.: (\exists x) \neg F x\)

2. \(A \cdot (B \cdot C), A \cdot E, B \cdot D \quad /.: (C \cdot D) \cdot E\)

3. \(B \supset (C \cdot L), C \supset (Q \lor T), (Q \lor T) \supset G, B \quad /.: G\)

4. \((\forall x) \neg P x \supset (\forall x) \neg T x, (\exists x) (T x \cdot Q x) \quad /.: (\exists x) (Q x \cdot P x)\)

5. The direction of this film was good, because it’s very success implies both that the marketing of the film was good and it had some popular appeal. That it had popular appeal entails that either the acting or the special effects were effective. If either the acting or the special effects were effective then the directing was good. The film was successful.

6. This argument is sound because an argument is sound provided it has only true premises and it is valid. The premises in this argument are all true. Furthermore, this argument is valid.

7. \((T \cdot M) \supset (B \supset A), T \cdot Z, (T \lor G) \supset (A \supset B), (A \supset B) \supset M \quad /.: A \equiv B\)

8. \(P \supset B, S \supset P, Q \supset C, C \supset S, Q \quad /.: B\)

9. \(A \lor B) \supset (C \supset D), M \supset (A \lor B), F \supset T, T \supset M, F \cdot C \quad /.: D\)

10. \((\forall x)[(F x \cdot G x) \equiv H x], (\forall x)(F x \supset G x) \quad /.: (\forall x)(F x \equiv H x)\)
B. More proofs

1. $(\forall x) (Lx \supset \sim Mx), (\forall x) [Sx \supset (Tx \supset Mx)], (\exists x) (Tx \cdot Sx) \quad /: \quad (\exists x) (Mx \cdot \sim Lx)

2. $(\exists x) (Bx \cdot Gx), (\forall x) [(Dx \cdot \sim Cx) \supset \sim Ax], (\forall x) [(Bx \supset \sim Ex) \equiv \sim Ax], (\forall x) (Bx \equiv Dx), (\forall x) [\sim Cx \equiv (Bx \cdot Dx)] \quad /: \quad (\exists x) (Gx \cdot \sim Ex)

3. Only Native Hawaiians who are citizens, may vote in OHA elections. John is a citizen, but not native Hawaiian. So, some citizens cannot vote in OHA elections.

4. $C \cdot L, A \equiv (B \cdot T), L \supset A, T \equiv (U \cdot V) \quad /: \quad B \cdot V$

5. $(M \lor N) \supset \sim O, P \supset (\sim O \cdot Q), Q \equiv [O \cdot (M \cdot R)] \quad /: \quad \sim P \equiv \sim Q$

6. $(L \lor T) \supset (S \cdot G) \quad /: \quad (L \cdot V) \supset (G \lor T)$

7. $(L \cdot M) \supset C, L \supset M, (D \lor C) \supset \sim G, (L \lor \sim G) \supset N \quad /: \quad N$

8. That which is due to chance does not reappear constantly nor frequently, but all products of Nature reappear constantly or at least frequently. Therefore, nothing in Nature is due to chance.

9. If we give generously to the campaign, the candidate will have an obligation to accept our bid on their next project. We can figure a more generous profit margin in preparing our estimates, if the candidate has an obligation to accept our bid on their next project. Figuring a more generous profit margin will cause our general financial condition to improve considerably. Therefore, considerable improvement in our general financial condition will follow from a generous donation to the campaign.

10. $\sim A \cdot (\sim B \supset C), (B \supset A) \cdot (C \supset \sim D), E \supset D, [(\sim A \cdot \sim B) \cdot \sim E] \supset F \quad /: \quad F$

11. Our judges have not acted well for these reasons: Judges have acted well only if all resistance to the law is morally wrong. But if civil disobedience is moral, then not all resistance to the law is morally wrong. But if all resistance to the law is morally wrong then our legal code is correct. Civil disobedience is moral if and only if either civil disobedience is moral or our legal code is correct.

12. $P \supset Q \quad /: \quad \sim (P \cdot \sim Q)$

13. Abe rents the apartment if Bla works overtime; however, if Chas can afford it, Dei will not move in. Abe will not rent the apartment, although, if Bla doesn’t work overtime, Chas will be able to afford it. Dei will move in provided Eli gets a raise. But if Abe doesn’t rent the apartment and Bla doesn’t work overtime, and Eli doesn’t get a raise, then Flo will return. So, Flo will return.
14. \( D \equiv (B \cdot C) \), \( A \equiv (\neg B \cdot \neg E) \)  \(/:\ (A \cdot D) \equiv F \)
15. \( B \equiv (T \cdot A) \), \( D \equiv T \), \( (A \vee D) \equiv B \), \( (B \equiv A) \equiv C \)  \(/:\ (A \cdot D) \equiv (C \cdot T) \)

16. \( (A \cdot B) \equiv (C \vee D) \), \( C \equiv E \), \( A \equiv B \), \( A \cdot F \), \( (D \equiv G) \cdot H \)  \(/:\ (A \cdot B) \cdot (E \vee G) \)
17. \( \neg A \equiv B \), \( \neg (A \cdot B) \)  \(/:\ A \)

18. \( (A \cdot B) \equiv C \), \( (B \equiv C) \equiv (B \vee D) \), \( \neg (D \vee E) \cdot B \)  \(/:\ \neg A \)
19. \( L \vee (M \vee T) \)  \(/:\ (M \vee L) \vee T \)

20. \( \forall x)(\exists y) Sxxy, (\forall x)(\forall y)(Sxxy \equiv Syx) \)  \(/:\ (\exists x)(\forall x) Sxxy \)
21. \( (\forall x)(C \equiv x \equiv Mx, Da, (\forall x)(\forall y)(Dxy \equiv Cxy) \)  \(/:\ \neg Ma \)

C: Practicing with assumptions

1. \( (F \vee G) \equiv [(H \vee I) \equiv K] \)  \(/:\ F \equiv (H \equiv K) \)
2. \( A \equiv C \)  \(/:\ (A \cdot B) \equiv (B \equiv C) \)

3. \( P \equiv (Q \equiv R) \), \( Q \)  \(/:\ \neg R \equiv \neg P \)
4. \( A \equiv (\neg B \cdot C) \), \( \neg B \equiv \neg C \)  \(/:\ A \equiv \neg D \)

5. \( P \equiv (Q \equiv R) \)  \(/:\ Q \equiv (P \equiv R) \)
6. \( T \equiv (U \cdot V) \), \( V \equiv (W \cdot L) \)  \(/:\ (A \equiv V) \equiv (T \equiv W) \)

7. \( A \equiv (B \cdot C) \), \( A \equiv (C \cdot D) \)  \(/:\ (B \equiv \neg D) \equiv A \)
8. \( F \equiv (\neg C \equiv \neg E) \), \( G \equiv [F \cdot (B \cdot E)] \)  \(/:\ G \equiv C \)

9. \( A \equiv B \), \( \neg A \)  \(/:\ B \)
10. \( (\forall x)(Gx \equiv Vx) \), \( (\exists x)(\neg Cx \equiv \neg Gx) \)  \(/:\ (\exists x) Vx \)

11. \( P \equiv Q \), \( (P \equiv R) \equiv S \)  \(/:\ P \equiv [(Q \equiv R) \equiv S] \)
12. \( M \equiv V \), \( (M \equiv R) \equiv \neg S \)  \(/:\ M \equiv [(V \equiv R) \equiv \neg S] \)

13. \( (C \equiv D) \cdot (E \equiv \neg D) \)  \(/:\ C \equiv \neg E \)
14. \( P \equiv Q \), \( \neg P \equiv Q \)  \(/:\ Q \)

15. \( (A \equiv B) \equiv \neg C \), \( C \equiv D \), \( A \equiv D \)  \(/:\ D \)
16. \( L \equiv (T \equiv F) \), \( \neg L \cdot (T \equiv G) \)  \(/:\ G \)
17. \( A \equiv (B \equiv C) \), \( (B \equiv C) \equiv D \)  \(/:\ A \equiv (B \equiv D) \)
18. \( E \vdash (F \supset G), \ H \vdash (G \supset I), \ (F \supset I) \vdash J \) \hfill \( \vdash (E \cdot H) \supset J \)

19. \( A \vdash (\neg C \supset \neg B), \ \neg C \vdash (A \cdot B) \) \hfill \( \vdash A \supset C \)

20. \( \neg (P \cdot \neg Q) \) \hfill \( \vdash P \supset Q \)

21. \( A \vdash B, \ \neg D \vdash \neg B, \ \neg A \vdash C, \ \neg D \lor E \) \hfill \( \vdash C \lor E \)

22. \( (E \cdot F) \lor (G \equiv H), \ I \equiv G, \ \neg (E \cdot F) \) \hfill \( \vdash I \equiv H \)

23. \( T \vdash U, \ V \lor \neg U, \ \neg V \cdot \neg W \) \hfill \( \vdash \neg T \)

24. \( N \vdash (Q \cdot P), \ Q \equiv (R \cdot S) \) \hfill \( \vdash (P \lor Q) \equiv (N \lor S) \)

25. \( A \vdash (B \equiv C), \ B \equiv (C \equiv D) \) \hfill \( \vdash A \equiv (B \equiv D) \)

26. \( J \lor \neg K, \ K \lor (L \equiv J), \neg J \) \hfill \( \vdash L \equiv J \)

27. \( (H \equiv I) \cdot (J \equiv K), \ K \lor H, \ \neg K \) \hfill \( \vdash I \equiv \neg J \)

28. \( A \equiv \{[B \equiv (C \equiv D)]\}, \ D \equiv (E \equiv F) \) \hfill \( \vdash (A \cdot B) \equiv [E \equiv (C \equiv F)] \)

29. \( (L \cdot S) \equiv (G \cdot H), \ L \equiv S, \ H \equiv G \) \hfill \( \vdash L \equiv H \)

30. \( (\forall x) [(Sx \lor Lx) \equiv Tx], \ (\forall x) (Lx \equiv Sx) \) \hfill \( \vdash (\forall x) [Sx \equiv (Lx \cdot Tx)] \)

31. \( (\forall x) (Ax \equiv Gx), \ (\forall x) [Dx \equiv (\neg Gx \cdot Cx)], \ (\forall x)[\neg Gx \equiv (Cx \equiv Dx)], \ (\forall x) Ax \) \hfill \( \vdash (\forall x)(Cx \equiv Dx) \)

32. \( (A \cdot B) \equiv C, \ (B \equiv C) \equiv (B \equiv D), \neg (D \lor E) \cdot B \) \hfill \( \vdash \neg A \)

33. \( P \equiv H, \ H \equiv (P \cdot Q) \) \hfill \( \vdash P \equiv (H \cdot Q) \)

34. \( (P \cdot Q) \equiv (R \equiv S) \) \hfill \( \vdash P \equiv [Q \equiv (R \equiv S)] \)

35. \( (G \cdot T) \lor (U \lor V), \ \neg U \lor \neg T \) \hfill \( \vdash V \)

36. \( A \equiv D, \ D \equiv (G \lor F), \ (G \equiv I) \cdot (F \equiv I), \ I \equiv B \) \hfill \( \vdash (A \cdot D) \equiv B \)

37. \( A \cdot B, \ A \equiv \neg C, \ B \equiv D, \ \neg C \lor H, \ A \equiv E \) \hfill \( \vdash (H \cdot D) \cdot E \)

38. \( D \equiv (B \cdot C), \ A \equiv (\neg B \cdot \neg E) \) \hfill \( \vdash (A \cdot D) \equiv F \)

39. \( (B \equiv A) \cdot \neg C, \ \neg C \equiv (D \cdot \neg A), \ B \lor E \) \hfill \( \vdash E \)

40. \( (T \cdot Y) \equiv Z, \ (A \lor X) \equiv T \) \hfill \( \vdash X \equiv [Y \equiv (\neg Z \equiv A)] \)

41. \( (A \lor B) \cdot \neg C, \ \neg C \equiv (D \cdot \neg A), \ B \equiv (A \lor E) \) \hfill \( \vdash E \lor F \)

72
D. Practice with Translation in Sentential

**Exercise:** First, determine the validity of each of the arguments below (using any method that is suitable to the type of argument). **Complete the argument context for any enthymemes.** Then, if the argument is valid, provide a proof that demonstrates this fact.

1. If concepts are not clear, words do not fit. If words do not fit, the day’s work cannot be accomplished, morals and art do not flourish. If morals and art do not flourish, punishments are not just. If punishments are not just, the people do not know where to put hand or foot.

   Confucius, *Analects*, XIII,3

2. If the jam is tasty then the berries are sweet. Either the berries are sweet or sugar must be added. If sugar must be added but the jam is not tasty, then the jam will not sell well. The berries are not sweet. We can conclude the jam will not sell well.

3. If the GNP has decreased for three consecutive quarters and layoffs are increasing, then the country is in a recession. Layoffs are increasing but the GNP has not decreased for three consecutive quarters. Therefore, the country is not in a recession.

4. If the teller or the cashier had triggered the alarm, the vault would have locked automatically and the police would have arrived within five minutes. Had the police arrived in five minutes, the robbers would not have gotten away. But they did get away. So, the teller did not trigger the alarm.

5. Either Jack or Jill is a member of the Student Congress. If either Jill or Randy is a member then Jack is not. Jack is not a member if, and only if both Jill and Randy are. Jack is a member. So, if Jill is a Student Congress member then Randy is not.

6. Either the governor’s assassination was the work of the mob, or else it was an inside job and the Secret Service is implicated. The mob committed the crime only if its assassins knew the exact location of the governor. But the Secret Service is implicated if the mob’s assassins knew the exact location of the governor. Therefore, the Secret Service is implicated.

7. If interest rates on bank loans increase, then mortgage money will become tighter and there will be fewer housing starts. There will be fewer housing starts only if personal income does not increase. Consequently, if interest rates on bank loans increase, personal income will not.

8. Either we have red wine or Primo for dinner. If we have steak we’ll have red wine, but if we have pizza, we’ll have Primo. We’ll have steak if and only if we have fruit salad. But we’ll have fruit salad if we have beer. Therefore, we won’t have pizza and Primo.
E. More Practice Proofs in Sentential:

1. \(P \supset (G \supset T), \ P \cdot Q, \ Q \supset (T \supset E)\) \(\vdash G \supset E\)
2. \(T \supset R, \ T \supset \neg R\) \(\vdash \neg T\)
3. \(C \supset (G \cdot M), \ \neg M\) \(\vdash \neg C\)
4. \(M \supset (K \supset L), \ (L \lor N) \supset J\) \(\vdash M \supset (K \supset J)\)
5. \(R \supset B, \ R \supset (B \supset F), \ B \supset (F \supset H)\) \(\vdash R \supset H\)
6. \(\neg M \supset (N \cdot O), \ N \supset P, \ O \supset \neg P\) \(\vdash \neg M\)
7. \(P \equiv [(L \lor M) \equiv (N \cdot O)], \ (O \lor T) \equiv W\) \(\vdash P \equiv (M \equiv W)\)
8. \(P \equiv (I \equiv W), \ I \equiv (W \equiv \neg S)\) \(\vdash P \equiv (I \equiv \neg S)\)
9. \((S \lor T) \equiv \neg S\) \(\vdash \neg S\)
10. \((K \lor L) \equiv (M \cdot N), \ (N \lor O) \equiv (P \cdot \neg K)\) \(\vdash \neg K\)
11. \((C \cdot R) \equiv (I \cdot D), \ R \equiv \neg D\) \(\vdash C \equiv \neg R\)
12. \(K \equiv [(M \lor N) \equiv (P \cdot Q)], \ L \equiv [(Q \lor R) \equiv (S \cdot \neg N)]\) \(\vdash (K \cdot L) \equiv \neg N\)
13. \(J \equiv (D \equiv C), \ (N \cdot C) \equiv I\) \(\vdash J \equiv (N \equiv I)\)
14. \(B \equiv (G \equiv F), \ (F \equiv N) \equiv (G \cdot N)\) \(\vdash \neg (B \equiv F)\)
15. \(F \equiv (G \cdot H)\) \(\vdash (A \equiv F) \equiv (A \equiv H)\)
16. \(P \equiv (Q \equiv R), \ \neg R\) \(\vdash P \equiv \neg Q\)
17. \(X \equiv Y, \ (Y \equiv \neg X) \equiv (Y \equiv Z)\) \(\vdash \neg Z \equiv \neg X\)
18. \(X \equiv Z, \ Y \equiv Z\) \(\vdash (X \equiv Y) \equiv Z\)
19. \(\neg A \cdot (B \equiv C), \ B \equiv C, \ C \equiv A\) \(\vdash D\)
20. \(\neg D\)
21. \(\neg D, \ (D \lor B) \equiv \neg A, \ (A \equiv B) \equiv (D \cdot E), \ \neg B \equiv D, \ C \equiv (A \equiv B)\) \(\vdash C \equiv (\neg A \cdot B)\)
22. \(Z \equiv (X \equiv Y), \ X \equiv \neg W, \ Y \equiv \neg W, \ \neg W \equiv \neg Z\) \(\vdash \neg Z\)
23. \(P \equiv Q, \ \neg P \equiv J, \ \neg Q \equiv \neg J\) \(\vdash Q\)
24. \(D \equiv (A \equiv B), \ C \equiv D\) \(\vdash A \equiv \neg C\)
25. \((A \equiv B) \equiv C, \ C \equiv (D \equiv A), \ B \equiv (A \equiv B)\) \(\vdash E \equiv F\)
26. \(F \equiv (H \equiv (L \equiv G), \ (G \cdot B) \equiv [(G \cdot (K \equiv G)])\) \(\vdash F\)
27. \((H \cdot T) \equiv J, \ (M \equiv D) \cdot (\neg D \equiv M), \ \neg T \equiv (\neg D \cdot M)\) \(\vdash H \equiv J\)
28. \(M \equiv A, \ [M \equiv (A \cdot M)] \equiv [C \equiv (A \cdot M)], \ \neg (A \cdot M) \equiv (C \equiv \neg D)\) \(\vdash \neg D\)
29. (R v W), (R = M) v [(M v G) ⊃ (W = M)]
30. B ⊃ (K • M), (B • M) ⊃ (P ⇔ P)

31. A ⊃ (B v C), D ⊃ ~ C
32. A ⊃ (Q • B), (~Q ⇔ B) • (C ⊃ A)

33. A ⊃ (B • C), ~B
34. (A v N) ⊃ ~ S, M • [N e (S ⊃ T)]

35. A • D, B v C, A ⊃ ~ B
36. (V v D) • J, D ~ X

37. ~ (C • ~ A)
38. D e (B @ C), H • A, A e C)

F. Extra practice with Predicate Rules:

1. (∀x) (A x ⊃ B x), (∀x) Ax
   /: (∀x) B x
2. (∀x) (~ Ax ⊃ ~ B x), (∃x) ~ Ax
   /: (∃x) ~ B x

3. ~ (∃x) B x, (∀x) ~ B x • (∀x) ~ C x
4. (∀x) Ax ⇔ (∃x) ~ B x, (∀x) G x • (∀x) A x
   /: ~ (∀x) B x

5. ~ (∃x) (A x • B x), (∃x) (A x • C x)
6. ~ (∀x) A x ⇔ (∃x) B x, (∃x) ~ A x • (∃x) C x
   /: (∀x) A x • (∃x) C x

7. ~ (∀x) (A x ⊃ B x), (∀x) (A x ⊃ ~ C x)
8. Fa • ~ Pa, (∀x) (F x ⊃ ~ R x), (∀x) (~ P x • T x)
   /: ~ Ra • Ta

9. (∀x) (Ax ⇔ B x), (∃x)(Ax • ~ C x)
10. (∃x)(Ax • B x), (∀x) (A x ⊃ C x), (∀x) (D x ⇔ B x)
   /: (∃x)(C x • D x)

11. (∀x) ~ Ax ⊃ (∀x) ~ B x, ~ A a
12. (∀x) (A x ⊃ C x), (∀x) (C x ⊃ B x)
   /: (∀x) (A x ⊃ B x)

G. Practice with translating

Translate the following arguments into either Sentential or Predicate. Use all necessary tools you have learned in this course, your knowledge and your common sense to determine if they are valid, sound, and persuasive.

1. The Saudi’s are truly partners in the fight against terrorism provided they are able to capture and prosecute terrorists in their own country, as well as curb radical teachings in their schools. They are able to capture some terrorists but they are not able to curb radical teachings in their schools. So, we can conclude that they are not true partners.
2. The presidential election will be won by either a Republican or a Democrat, or it will be won by an Independent or a Libertarian. It will be won by a Republican only if McCain wins. It will be won by Democrat provided Obama wins. It will be won by an Independent if and only if Nadar wins. And a Libertarian winning implies Bob Bar wins. McCain will win only if Russia invades Poland, and Obama will win only if Americans can overcome their zenophobia. Barr needs name recognition and he won’t get it. So, Nadar will be our next president.

3. If John wants to finish his degree, he will need to take 12 credits each semester for the next year. If he is to take that many credits he will need to quit his job. John wants to finish his degree, so, John will need to have his parents co-sign a loan because if he is to quit his job, he will need to take out a loan. And if he takes out a loan he will need his parents to co-sign it.

4. Lying is moral because everyone lies. And if everyone lies then lying must be moral.

5. If low rain fall continues we will have to begin water restrictions since unrestricted water use can continue only if we fail to get greater amounts of rain.

6. Only valid arguments are sound. Any sound argument must have true premises. Whatever has true premises is persuasive. Some valid arguments are not persuasive. Some valid arguments are not persuasive.

7. The argument is valid or it is invalid. If it is invalid it cannot be sound. If it is not sound then it cannot be a good argument. Therefore, if the argument is both valid and sound, it is a good argument.

8. All valid arguments are consistent or inconsistent. No inconsistent can be sound. Any sound argument it must be both valid and consistent. So, an argument that is invalid is inconsistent.

More Practice: Analyze and evaluate fully the following arguments using ANY of the tools, concepts, strategies and techniques that you find applicable. Given the specific statements in the argument, make a judgment about the arguments’ soundness, defending your judgment with specific reference to the argument’s structure or content. Supply missing statements for any enthymemes.

1. Some teachers are not well paid. We know this for certain because anyone who is really well paid drives a nice car and there are teachers who do not drive nice cars.
2. Either I take Logic or I take Math. I’m taking Logic. So I won’t be taking math.

3. We will not elect a woman as president of the US in 2008 because, we will only elect a woman if we vote Green and Americans are not ready to vote Green.

4. The air force has a saying, “If you not catching flack, you’re not over the target. I’m catching flack, so I must be over the target.” Mike Huckabee, Presidential Candidate, Republican Debate 1/10, 2008

5. If we want to have a woman president we can elect Hillary Clinton or Cynthia McKinney. If we want a minority president, we can elect Barack Obama or Cynthia McKinney. We want to elect both a woman and a minority. So, we should elect Cynthia McKinney.

6. If I go to college, I’ll never achieve the American Dream. Here’s the reasoning: If I go to college, I can’t work. If I can’t work, I can’t save money. If I can’t save money, I can’t buy a home. If I can’t buy a home, I’ll never achieve the American Dream.

7. If DARE were a successful drug program then kids who graduate from it in the sixth grade would not do drugs when they get into high school. Many kids who do graduate from DARE in the sixth grade end up doing drugs in high school. So, DARE cannot be that successful.

8. If the dollar continues to weaken, goods from China will become more expensive and US consumers will not buy them any more. If US consumers don’t buy Chinese goods, China will fall into recession and will not want to continue to buy our debt. The dollar will not continue to weaken. So, China will continue to want to buy our debt.

9. If I don’t need my respirator on New Years Eve, then either the trades will be strong or there will be fewer fireworks than usual. If there are fewer fireworks than usual, fireworks sales will be low. If the trades are strong there will be no haze. There is a haze and fireworks sales have been strong. So, I’ll need my respirator.

10. All registered students are eligible. No faculty is invited. Everyone here is either ineligible or invited. Therefore, everyone here is either not a student or not a faculty member.

The following arguments come from the writings of ancient philosophers in the Greek and Roman traditions. They illustrate how deductive logic was used to present arguments on a variety of topics. Consider their validity, their soundness and their persuasiveness.

1. That which is evil does harm; that which does harm makes a man worse. But pain and poverty do not make a man worse; therefore, they are not evil.

   Seneca

2. Death is one of two things. Either it is annihilation, and the dead have no consciousness of anything, or . . . it is really a change – a migration of the soul to another place. Now if there were no consciousness . . . death must be a marvelous gain. . . . If on the other hand death is a removal from here to some other place . . ., what a greater blessing could there be than this.
Plato, *The Apology*

3. Either the boy has at some time acquired knowledge (of geometry) which he now has, or he has always possessed it. If he always possessed it, he must always have known. If on the other hand he acquired it some time previously, it cannot have been in this life, unless somebody has taught him geometry. . . . Has anyone taught him these things?

   I know that no one ever taught him. . . .

   Then if he did not acquire them in this life, isn’t it immediately clear that he possessed and learned them during some other period. 

   Plato, *The Meno*

4. Concerning these things the decision lies here: either it is or it is not. But it has been decided . . . that the one way (that it is not) is unknowable and unnameable. . . . For if it came into being, it is not; nor is it if ever it is going to be. Thus coming into being (is not possible.)

   Parmenides

5. If being is, it must be one. On the other hand, if it is one, it cannot have body. [For if it had body it would have thickness.] If it had thickness it would have parts, and [if it had parts it] would no longer be one.

   Melissus

7. Zeno argues thus: Either a moving object moves in the place where it is, or in the place where it is not. And it does not move in the place where it is, nor in the place where it is not; therefore nothing moves.

8. Teaching does not come by ocular evidence, since ocular evidence consists in things exhibited. But what is exhibited is apparent to all; and the apparent, . . . is perceptible by all; and what is perceptible by all in common is incapable of being taught; therefore nothing is capable of being taught by ocular evidence.

   Sextus Empiricus

9. There is good only in pleasure and bad only in pain. There is neither pleasure nor pain in death. Therefore, death is neither good nor bad.
Answers to Selected Exercises:

Answers to odd exercises p. 13. Truth Tables for Validity

1. \( p \rightarrow q \)

\[
\begin{array}{c|c|c}
P2 & p & \rightarrow q \\
\hline
T & T & T \\
T & F & F \\
F & T & T \\
F & F & T \\
\end{array}
\]

This argument is valid. There is no row of false conclusion where all premises are true.

3. \( p \rightarrow q \)

\[
\begin{array}{c|c|c}
p & \rightarrow q & P1 \ p \rightarrow q \\
\hline
T & T & T \\
T & F & F \\
F & T & T \\
F & F & T \\
\end{array}
\]

This argument is invalid by row 4 which shows a false conclusion and a true premise.

5. \( p \lor q \)

\[
\begin{array}{c|c|c|c|c}
p & \lor q & \sim p & P1 \ p \lor q \\
\hline
T & T & F & T \\
T & F & F & F \\
F & T & T & T \\
F & F & F & T \\
\end{array}
\]

This argument is valid. There is no row of false conclusion where all premises are true.

7. \( p \lor q \)

\[
\begin{array}{c|c|c|c|c}
p & \lor q & P1 \ p \lor q \\
\hline
T & T & T & T \\
T & F & F & F \\
F & T & T & T \\
F & F & T & T \\
\end{array}
\]

This argument is invalid by row 3, which shows a false conclusion and a true premise.

9. \( p \cdot q \)

\[
\begin{array}{c|c|c|c|c}
p & \cdot q & P1 \ p \cdot q \\
\hline
T & T & T & T \\
T & F & F & F \\
F & T & F & F \\
F & F & F & F \\
\end{array}
\]

This argument is valid.

11. \( p \cdot \sim p \)

\[
\begin{array}{c|c|c|c|c}
p & \sim p & \cdot q & P1 \ p \cdot \sim p \\
\hline
T & F & T & F \\
T & F & F & F \\
F & T & T & F \\
F & T & F & F \\
\end{array}
\]

This argument is valid. There is no row of false conclusion where all premises are true.
13. \( p \equiv q \)
   \( \sim p \sim q \)

\[
\begin{array}{c|c|c|c|c}
 p & q & P1 p = q & P2 \sim p & \therefore \sim q \\
 T & F & T & F & F \\
 T & F & F & F & T \\
 F & T & F & T & F \\
 F & T & T & T & T \\
\end{array}
\]

This argument is valid. There is no row of false conclusion where all premises are true.

15. \( p \supset q \)
   \( q \supset p \)
   \( \therefore p \equiv q \)

\[
\begin{array}{c|c|c|c|c}
p & q & P1 p \supset q & P2 q \supset p & \therefore p \equiv q \\
T & F & T & T & T \\
T & F & F & T & F \\
F & T & T & F & F \\
F & T & T & T & T \\
\end{array}
\]

This argument is valid. There is no row of false conclusion where all premises are true.

Answers to 1 and 5 p. 15: Tautology test

1.

\[
\begin{array}{c|c|c|c}
(A \supset B) & \cdot & \sim B & \supset A \\
T & F & F & T \\
T & F & T & T \\
F & F & T & T \\
T & F & F & T \\
T & T & T & F \\
\end{array}
\]

The statement is contingent. Therefore, the argument is invalid.

5.

\[
\begin{array}{c|c|c|c|c|c|c|c|c|c|c}
\{A \vee B\} & \cdot & (B \supset C) & \cdot & C & \supset & \sim A \\
T & T & T & T & T & F & F \\
T & T & F & F & F & T & F \\
T & T & T & T & T & F & F \\
T & T & T & F & F & T & F \\
T & T & T & T & T & T & T \\
T & T & F & F & F & T & T \\
T & F & T & F & T & T & T \\
T & F & T & F & T & T & T \\
\end{array}
\]

The statement is contingent. Therefore, the argument is invalid.
The answers to * exercises p. 18

**Argument check column goal grid**

1. \( P \equiv Q \)
   \( T \equiv T \)
   \( T \)
   \( P \quad Q \quad R \)

\( \sim Q \lor R \)
   \( \sim T \lor T \)
   \( T \)
   \( T \quad T \quad T \)

\( \therefore R \supset \sim P \)
   \( T \supset \sim T \)
   \( F \)

This argument is invalid.

**Argument check columns goal grid**

3. \( \sim (H \cdot \sim I) \)
   \( \sim (F \cdot \sim T) \)
   \( \sim (F) \)
   \( T \quad H \quad I \quad J \quad K \)

\( I \lor (J = K) \)
   \( T \lor (T = T) \)
   \( T \lor T \)
   \( T \quad F \quad T \quad T \quad T \)

\( \therefore \sim (J \cdot I) \)
   \( \sim (T \cdot T) \)
   \( \sim T \)
   \( F \)

This argument is invalid.

---

**Answers to odds p. 23**

1. \( T \lor R, T \supset \sim R, R \)
   \( \therefore T \)
   Consistent / Unsound because invalid

3. \( A \supset \sim A \)
   \( \therefore \sim A \lor \sim A \)
   Consistent / Unsound because invalid

5. \( A \supset (B \cdot C), \sim B \)
   \( \therefore A \supset C \)
   Consistent / Unsound because invalid

7. \( A \lor D, B \lor C, A \supset \sim B \)
   \( \therefore C \lor D \)
   Consistent / Possibly sound be/valid & consistent

---

**Answers to odd exercises p.28**

1. \( Qa \)
11. \( Ga \supset Gi \lor (\forall x) Gx \supset (\exists x) Gx \)
3. \( Sa \lor \sim Ta \)
13. \( Ba \supset (\exists x) Bx \)
5. \( Ra \supset \sim Fa \)
15. \( (\exists x) \sim Jx \)
7. \( (\exists x)(Mx \cdot \sim Tx) \)
17. \( (\exists x)(\sim Jx \cdot \sim Mx) \)
9. \( (\exists x) Sx \lor F1 \lor Ss v (\exists x) Fx \)
19. \( (\exists x) [(Gx \cdot Lx) \lor \sim Kx] \)

**Answers to odd exercises p. 30**

1. \( (\forall x) Ax \)
3. \( (\forall x) (Gx \lor \sim Cx) \)
5. \( (\forall x) \sim (Mx \cdot Vx) \)
7. \( (\forall x)(\forall x) Axx \)
9. \( (\forall y)(\forall x)(Fx \cdot \sim Sxy) \)
11. \( (\forall x) [R \supset (Dx \cdot Tx)] \)
Answers to odd exercises (5-15) p. 33.

5. $(\exists x) \sim Bx \quad \sim (\forall x) \sim Bx$

7. $\sim (\forall x) Bx \quad (\exists x) \sim Bx$

9. $\sim (\exists x) (Bx \cdot Ax) \quad (\forall x) (Bx \Rightarrow \sim Ax)$

11. $(\exists x) (Bx \cdot \sim Ax) \quad \sim (\forall x) (Bx \Rightarrow Ax)$

13. $\sim (\forall x) (Ax \Rightarrow \sim Bx) \quad (\exists x) (Ax \cdot \sim \sim Bx)$

15. $(\exists x) \sim Bx \Rightarrow (\forall x) Ax \quad \sim (\forall x) Bx \Rightarrow (\forall x) Ax$ Using LE on the antecedent

OR $(\exists x) \sim Bx \Rightarrow (\exists x) \sim Ax$ Using LE on the consequent

---

Partial Answers to odd exercises pp 38-39: These answers present the derivation column. You should fill in justification column yourself by looking for where a change occurs in each subsequent line. Then identify the rule that allows that change.

1.
1. $(\forall x) N x$  $/\vdash (\exists x) N x$
2. Na
3. $(\exists x) N x$

This proof presents the basic steps required to take any universally quantified statement into an existentially quantified statement. Notice how this pattern plays through in other situations below.

3.
1. $\sim (\exists x) S x \cdot (\forall x) M x$  $/\vdash (\exists x) (M x \cdot \sim S x)$
2. $(\forall x) \sim S x \cdot (\forall x) M x$
3. $\sim S a \cdot (\forall x) M x$
4. $\sim S a \cdot M a$
5. $\sim S a$
6. Ma
7. Ma $\cdot \sim S a$
8. $(\exists x) (M x \cdot \sim S x)$
5.
1. \((\forall x) (N_x \cdot T_x)\)  
2. \((\forall x) (M_x \cdot S_x)\)  
3. \(N_a \cdot T_a\)  
4. \(M_a \cdot S_a\)  
5. \(N_b\)  
6. \(S_a\)  
7. \(N_a \cdot S_a\)  
8. \((\exists x) (N_x \cdot S_x)\)

7.
1. \((\forall x) (N_x \cdot T_x)\)  
2. \((\exists x) (S_x \cdot \neg L_x)\)  
3. \(S_t \cdot \neg L_t\)  
4. \(N_t \cdot T_t\)  
5. \(T_t\)  
6. \(\neg L_t\)  
7. \(T_t \cdot \neg L_t\)  
8. \((\exists x) (T_x \cdot \neg L_x)\)  

9.
1. \(\neg (\exists x) N_x\)  
2. \(\neg (\exists x) S_x\)  
3. \((\forall x) \neg N_x\)  
4. \((\forall x) \neg S_x\)  
5. \(\neg N_r\)  
6. \(\neg S_r\)  
7. \(\neg N_r \cdot \neg S_r\)  
8. \((\forall x) (\neg N_x \cdot \neg S_x)\)
Exercises: Proofs Using Conditional Elimination Rules/ odds p 44

1. Proof:
1. \( \sim (\exists x) (Lx \cdot Tx) \)
2. \((\forall x) Lx \) \( \therefore (\forall x) \sim Tx \)
3. \((\forall x)(Lx \sim Tx) \) LE 1
4. \( Lr \) \( \forall E 2 \)
5. \( Lr \sim Tr \) \( \forall E 3 \)
6. \( \sim Tr \) \( \Rightarrow E 4,5 \)
7. \((\forall x) \sim Tx \) \( \forall I 6 \)

Why was “r” used in this \( \forall E \) move?

What makes it possible to correctly apply \( \forall I \) rule?

3. Proof:
1. \((\forall x) (Sx \Rightarrow Kx) \)
2. \( Ss \) \( \therefore Ks \)
3. \( Ss \Rightarrow Ks \) \( \forall E 1 \)
4. \( Ks \) \( \Rightarrow E, 2, 3 \)

For the following (5-17 odds), the left hand column is completed. You need to provide the appropriate justification for all moves after the given premises.
5. **Proof:**
1. \( A \cdot (B \cdot \sim C) \)
2. \( A \supset (B \supset D) \) 
\( \therefore \) \( D \cdot \sim C \)
3. \( A \)
4. \( B \cdot \sim C \)
5. \( B \)
6. \( \sim C \)
7. \( B \supset D \)
8. \( D \)
9. \( D \cdot \sim C \)

7. **Proof:**
1. \( P \supset Q \)
2. \( P \supset G \)
3. \( M \cdot P \) 
\( \therefore \) \( (Q \cdot G) \cdot M \)
4. \( M \)
5. \( P \)
6. \( Q \)
7. \( G \)
8. \( Q \cdot G \)
9. \( (Q \cdot G) \cdot M \)

9. **Proof:**
1. \( (\forall x) \ Ux \)
2. \( (\forall x) \ (Ux \supset Gx) \) 
\( \therefore \) \( (\exists x)Gx \)
3. \( Ua \)
4. \( Ua \supset Ga \)
5. \( Ga \)
6. \( (\exists x)Gx \)
11. **Proof:**
1. \( \sim (\exists x) (Gx \cdot Tx) \)
2. \((\forall x) Gx \) /: \((\forall x) \sim Tx\)
3. \((\forall x) (Gx \supset \sim Tx)\)
4. Gr \supset \sim Tr
5. Gr
6. \sim Tr
7. \((\forall x) \sim Tx\)

What changes are made on line 1?

13. **Proof:**
1. \( \sim (\forall x) Ux \supset (\exists x) \sim Cx \)
2. \((\exists x) \sim Ux \) /: \(\sim (\forall x) Cx\)
3. \((\exists x) \sim Ux \) \(\supset (\exists x) \sim Cx\) LE, 1 NOTE: all logical equivalence rules may be done on a part of a line
4. \((\exists x) \sim Cx\)
5. \(\sim (\forall x) Cx\)

There is another sequence that will work in this proof. Can you think of it?

15. **Proof:**
1. \( (P \supset B) \supset [O \supset (T \supset W)]\)
2. \((P \supset B) \cdot (O \cdot T) \) /: \(W\)
3. \(P \supset B\)
4. \(O \supset (T \supset W)\)
5. \(O \cdot T\)
6. \(O\)
7. \(T \supset W\)
8. \(T\)
9. \(W\)

NOTE: There are sequences for this proof. Can you think of one?
17. **Proof:**
1. M · Z
2. (Z · ~H) ⊃ [J ⊃ (O · Y)]
3. J · ~H
4. Z
5. ~H
6. J
7. J ⊃ (O · Y)
8. J
9. O · Y
10. O
11. Y
12. Y · O

For 19 and 21, the Right hand column is provide. You need to provide the derivations that correspond. Other valid sequences are possible.

**19. Proof:**
1. (∀x) (Ax ⊃ ~Bx)
2. (∃x) (Ax ⊢ ~Cx) /: (∃x)(~Bx · ~Cx)
3. ∃E 2
4. ∀E 1
5. ·E 3
6. ·E 3
7. ⊃E 4, 5
8. ⊃I 6, 7
9. ∃I 8

**21. Proof:**
1. ~ (∃x) (Ax · Cx)
2. (∀x) (~Cx ⊃ ~Bx)
3. Aa /: ~Ba
4. LE 1
5. ∀E 4
6. ∀E 2
7. ⊃E 5,3
8. ⊃E 6, 7
Answers to odd exercise p. 46-47:  CONDITIONAL INTRODUCTION RULE

Develop schema for the following problems. Then use the schema to construct the complete proof.

1. N ⊨ M,  M ⊨ T  /: N ⊨ T

Proof Schema

Proof:
→ N  assumption
.
.
 n T  subgoal
n+1  N ⊨ T  ⊨I

3. For this next problem, “stripping off” the quantifiers and noting that the conclusion is universally quantified lets you know this proof requires the use of “r.” Set the schema up using “r” as the individual.

(∀x)(Qx ⊨ Fx), (∀x)[ Lx ⊨ (Wx ⊨ Qx)]  /: (∀x) (Lx ⊨ Fx)

Proof Schema:

Proof:
→ Lr  assumption
.
Fr  subgoal
Lr  ⊨ Fr  ⊨I
5. For this next problem, before you do a schema, consider what individual constant will replace the variable and base your schema on that individual.

\((\forall x) [Ax \supset (Bx \cdot Cx)], (\forall x)(Bx \supset Ex)\)  \(/:\ (\forall x) (Ax \supset Ex)\)

Proof Schema Proof:

\[\rightarrow Ar\] assumption
\[\rightarrow Er\] Subgoal
\[Ar \supset Er \supset I\]

For the following problems complete the justification column for each proof. The derivation column has been done for you.

7. Proof:
1. \(F \supset (I \supset H)\)
2. \(I \supset (H \supset J)\)
3. \(J \supset (I \supset K)\)  \(/:\ (F \cdot I) \supset K\)
4. \(F \cdot I\) assumption
5. \(F\)
6. \(I\)
7. \(I \supset H\)
8. \(H \supset J\)
9. \(H\)
10. \(J\)
11. \(I \supset K\)
12. \(K\)
13. \((F \cdot I) \supset K\)
9. Proof:
1. \((\forall x) \, Gx \vdash Gi\)
2. \(Gi \vdash El\)
3. \(El \vdash (\forall x) \, Ax\) \hfill \therefore (\forall x)Gx \vdash (\forall x)Ax
ger 4. \((\forall x) \, Gx\)
5. \(Gi\)
6. \(El\)
7. \((\forall x) \, Ax\)
8. \((\forall x) \, Gx \vdash (\forall x) \, Ax\)

11. Proof:
1. \(A \vdash (B \vdash C)\)
2. \(B \vdash [(E \cdot C) \vdash D]\) \hfill \therefore (A \cdot E) \vdash (B \vdash D)
ger 3. \((A \cdot E)\)
4. \(A\)
5. \(E\)
ger 6. \(B\)
7. \(B \vdash C\)
8. \(C\)
9. \(E \cdot C\)
10. \(D\)
11. \(B \vdash D\)
12. \((A \cdot E) \vdash (B \vdash D)\)

13. Proof:
1. \((\exists x)[ \, Px \vdash (Qx \vdash Rx)\] \hfill \therefore (\exists x) \, [Qx \vdash (Px \vdash Rx)]
ger 2. \(Pt \vdash (Qt \vdash Rt)\)
ger 3. \(Qt\)
ger 4. \(Pt\)
5. \(Qt \vdash Rt\)
6. \(Rt\)
9. \(Pt \vdash Rt\)
10. \(Qt \vdash (Pt \vdash Rt)\)
11. \((\exists x) \, [Qx \vdash (Px \vdash Rx)]\)
15. **Proof:**

1. \((\forall x)(Ax \supset Bx)\) \quad /: \quad (\forall x)Ax \supset (\forall x) Bx
2. Ar \supset Br
   \[\Rightarrow 3. (\forall x)Ax\]
   4. Ar
   5. Br
   \[\Rightarrow 6. (\forall x) Bx\]
7. \((\forall x)Ax \supset (\forall x) Bx\)

**Exercise p. 47:** Translation arguments. Remember the sequence of analysis: Separate statements, identify premises and conclusion, set up a language key, and translate into symbolic language. The arguments are analyzed and translated. The proofs are left for you to do.

1. If Holly goes, then Mary goes. If Jane goes than Jill goes. If Mary goes then Jane goes. So, it is certain that if Holly goes, Jill goes.

**Language Key:** Let H = Holly goes.
   M = Mary goes.
   A = Jane goes.
   J = Jill goes.

**Argument analysis:**

| Premise 1: If Holly goes, then Mary goes. | H \supset M |
| Premise 2: If Jane goes than Jill goes.  | A \supset J |
| Premise 3: If Mary goes then Jane goes.  | M \supset A |
| Conclusion: [If Holly goes, Jill goes ] | \therefore H \supset J |

Note: You can use Predicate just as easily in this problem. If you made this choice the language key would look be something like:
Let Gx = x goes
   h = Holly, m = Mary, j= Jane and l = Jill
3. All terrorists have known connections to Al Qaida. All those with known connections to Al Qaida come from Muslim countries or have Muslim connections. Therefore, all terrorists come from Muslim countries or have Muslim connections.

**Language Key:** Let $T_x = x$ is a terrorist  
$K_x = x$ has known connections to al Qaida  
$M_x = x$ comes from a Muslim country or has Muslim connections

**Argument Analysis:**
Premise 1: All terrorists have known connections to Al Qaida.
Premise 2: All those with known connections to Al Qaida come from Muslim countries or have Muslim connections
Conclusion: [A]ll terrorists come from Muslim countries or have Muslim connections.

**Argument Translation:**

$$(\forall x)(T_x \Rightarrow K_x)$$
$$(\forall x)(K_x \Rightarrow M_x) \quad \therefore (\forall x)(T_x \Rightarrow M_x)$$

**Exercise pp51-52. Proofs using BI-CONDITIONAL RULES**

1. **Proof:**
   1. $Q \equiv P$
   2. $M \Rightarrow (P \cdot C)$
   3. $M \cdot F$  
   $\quad \therefore Q$
   4. $M$
   5. $P \cdot C$
   6. $P$
   7. $Q$
3. Proof:
1. S · G
2. G \equiv (F \supset Q)
3. F
   \therefore Q · S
4. S
5. G
6. F \supset Q
7. Q
8. Q · S

4. X \supset (Y · Z), Z \supset (X · W)
   \therefore X \equiv Z

Proof Analysis: This proof requires \equiv I Rule. Therefore, analyze the conclusion into its 2 conditional statements. X \supset Z And Z \supset X. Since you cannot find either of these conditionals in the premises, you need to prove BOTH using \supset I rule.

Proof Schema:
Task One Task Two
\rightarrow X \rightarrow Z

\begin{align*}
\quad & . \\
\quad & . \\
\quad & Z \quad \text{Subgoal} \\
\quad & X \supset Z \supset I \\
\quad & X \supset X \supset I
\end{align*}

Proof:
1. X \supset (Y · Z)
2. Z \supset (X · W)
   \therefore X \equiv Z
   \rightarrow 3. X
   4. Y · Z
   5. Z
6. X \supset Z
   \rightarrow 7. Z
6. X \supset Z
   \rightarrow 7. Z
8. X · W
   9. X
10. Z \supset X
11. X \equiv Z
For exercises 5, 7 and 9, the proofs are NOT given. Challenge your thinking and skill level by using schema and “hints” to develop your own solutions,

5. \((A \lor C) \cdot M, (F \supset K) \equiv (A \lor C), T \cdot F \quad \therefore K \cdot M\)

This is a simple proof. Try it yourself! Hint: Divide and conquer. Look for the simple solution 1\textsuperscript{st} – when in doubt–start with the simplest move you can make. The simplest moves are immediate inferences, moves you can make directly off a single statement. Think: what does any conjunction imply about its conjuncts?

7. \(P \supset B, S \supset P, Q \supset C, C \equiv S, Q \quad \therefore B\)

This is also a simple proof. Given the Q is true, what derivation can you begin with? Then think: if I have derived it, can I use it?

9. \((\forall x)[(C x \supset Dx) \equiv (Z x \cdot L x)], (\forall x)(L x \equiv Tx), (\forall x)[(D x \supset C x) \equiv Tx], (\forall x)(L x \cdot Z x)\)

\[\therefore (\forall x)(D x \equiv C x)\]

Hint: Analyze the conclusion into its component conditionals. Can you find these in the premises? After stripping the quantifiers what is the easiest move to make? This one is meant to be a challenge.

11. \((L \equiv \sim M) \cdot (L \supset T), T \supset \sim M \quad \therefore \sim M \equiv (L \cdot T)\)

Hint: Make a schema by analyzing the conclusion into its component conditionals.

**Exercises p. 55: Working with Disjunctions**

1. **Proof:**
2. 1. \((K \lor O) \supset (G \cdot E)\)
3. 2. \(K \quad \therefore G\)
4. 3. \(K \lor O \quad \lor I \quad 2\)
5. 4. \(G \cdot E \quad \supset E, 1, 3\)
6. 5. \(G \quad \cdot E 4\)
3. Proof:
1. \((\exists x) \, Ax\)
2. \((\forall x) \, (Ax \supset Bx)\)
3. \((\forall x) \, [(Bx \lor Cx) \supset Dx] \therefore (\exists x) (Dx \cdot Bx)\)
4. \(At\)  \(\exists E\) 1
5. \(At \supset Bt\)  \(\forall E\) 2
6. \(Bt\)  \(\supset E\) 4,5
7. \(Bt \lor Ct\)  \(\lor I\) 6
8. \((Bt \lor Ct) \supset Dt\)  \(\forall E\) 3
9. \(Dt\)  \(\supset E\) 7,8
10. \(Dt \cdot Bt\)  \(\cdot I\) 6,7
11. \((\exists x) (Dx \cdot Bx)\)  \(\exists I\) 10

1. The argument below is a classic constructive dilemma proof. Study the schema to understand its pattern and apply the same pattern to variations on the theme.

   1. C \supset D
   2. E \supset F
   3. C \lor E \therefore D \lor F

   Schema:
   \(\rightarrow\) C
   \(\therefore\) D \lor F

   \(\rightarrow\) E
   \(\therefore\) D \lor F
   D \lor F \lor E

3. The argument below presents a similar pattern of reasoning. Note the similarities and differences.
1. \((E \lor F) \Rightarrow (C \cdot D)\)
2. \((D \lor G) \Rightarrow H\)
3. \(E \lor G\) \(\lor\) \(H\)

Schema:

\[\begin{align*}
\rightarrow E \\
\cdot \\
\cdot \\
\underline{H} \\
\rightarrow G \\
\cdot \\
\cdot \\
\underline{H} \\
H \lor E
\end{align*}\]

5. In this argument begin by removing the quantifiers to know what assumptions you will need. The mix of quantifiers tells you that you need to observe the restriction on \(\exists E\) rule.

\((\exists x) (Vx \lor Wx)\)
\((\forall x) (Zx \equiv Wx)\) \(\lor\) \(:\exists x(Zx \lor Vx)\)

Schema:

\[\begin{align*}
\rightarrow Vt \\
\cdot \\
\underline{Zt} \lor Vt \\
\rightarrow Wt \\
\cdot \\
\cdot \\
\underline{Zt} \lor Vt \\
Zt \lor Vt \lor E
\end{align*}\]

7. The derivation column is completed. You should be able to complete the justification column.
1. \((\forall x)(Gx \equiv Mx)\)
2. \((\exists x)(Gx \vee Mx)\)
3. \((\forall x) [ (Gx \supset (Mx \supset Tx)] \quad /:(\exists x)Tx\)
4. \(Gt \vee Mt\)
5. \(Gt = Mt\)
6. \(Gt \supset (Mt \supset Tt)\)

\(\Rightarrow\) 7. \(Gt\)
8. \(Mt\)
9. \(Mt \supset Tt\)
10. \(Tt\)

\(\Rightarrow\) 11. \(Mt\)
12. \(Gt\)
13. \(Mt \supset Tt\)
14. \(Tt\)
15. \(Tt\)
16. \((\exists x)Tx\)

9. In this problem the justification column is completed. Provide the corresponding derivations. Note that there are other sequences of steps that would be equally valid.

1. \((\forall x) (Xx \supset Yx) \cdot (\forall x) (Zx \supset \sim Wx)\)
2. \((\forall x) (Xx \vee Zx) \quad /:(\exists x) (Yx \vee \sim Wx)\)
3. \(\cdot E 1\)
4. \(\cdot E 1\)
5. \(\forall E ____\)
6. \(\forall E ____\)
7. \(\forall E ____\)

\(\Rightarrow\) 8. Assumption for \(\forall E\)
9. \(\supset E ____\)
10. \(\vee I \quad 9\)

\(\Rightarrow\) 11. Assumption for \(\forall E\)
12. \(\supset E ____\)
13. \(\vee I \quad 12\)
14. \(\vee E 7, 8-10, 11-13\)
15. \(\exists I 14\)
Exercise p. 57: Negation Introduction/Elimination

1. proof:
   1. ($\forall x)(Lx \supset Mx)$
   2. ($\exists x) \sim Mx$ $\therefore (\exists x) \sim Lx$
   3. $\sim Mt$
   4. $Lt \supset Mt$
      $\Rightarrow$ 5. $Lt$
      6. $Mt$
      7. $Mt \cdot \sim Mt$
   8. $\sim Lt$
   9. $(\exists x) \sim Lx$

3. proof:
   1. $\sim A \supset B$
   2. $B \supset A$ $\therefore A$
      $\Rightarrow$ 3. $\sim A$
      4. $B$
      5. $A$
      6. $A \cdot \sim A$
      7. $A$

5. proof:
   1. $B \supset D$
   2. $D \supset E$ $\therefore (B \cdot \sim E)$
      $\Rightarrow$ 3. $B \cdot \sim E$
      4. $B$
      5. $D$
      6. $E$
      7. $\sim E$
      8. $E \cdot \sim E$
      9. $\sim (B \cdot \sim E)$

No further answers will be given. All the basic strategies have been illustrated. Facility with proofs requires practice, persistence, insight and imagination. At this point you need to challenge yourself to build your skill and confidence.