DERIVATIONS IN SENTENTIAL LOGIC

1. Introduction ................................ ................................ ............................. 142
2. The Basic Idea................................ ................................ .......................... 143
3. Argument Forms and Substitution Instances................................. 145
4. Simple Inference Rules................................ ................................ ............. 147
5. Simple Derivations ............................................................................. 151
6. The Official Inference Rules................................ ..................................... 154
   • Inference Rules (Initial Set)................................ ................................ ...... 155
   • Inference Rules; Official Formulation ................................ ...................... 156
7. Show-Lines and Show-Rules; Direct Derivation .............................. 158
8. Examples of Direct Derivations................................ ................................ 161
9. Conditional Derivation........................................................................ 165
10. Indirect Derivation (First Form).......................................................... 170
11. Indirect Derivation (Second Form)...................................................... 175
12. Showing Disjunctions Using Indirect Derivation ............................. 178
13. Further Rules ..................................................................................... 181
14. Showing Conjunctions and Biconditionals................................. 182
15. The Wedge-Out Strategy ................................ ................................ .......... 185
16. The Arrow-Out Strategy ................................................................... 188
17. Summary of the System Rules for System SL............................... 190
18. Pictorial Summary of the Rules of System SL................................ 192
19. Pictorial Summary of Strategies............................................................. 196
20. Exercises for Chapter 5 ................................ ................................ ............ 199
21. Answers to Exercises for Chapter 5 .................................................. 204
1. INTRODUCTION

In an earlier chapter, we studied a method of deciding whether an argument form of sentential logic is valid or invalid – the method of truth-tables. Although this method is infallible (when applied correctly), in many instances it can be tedious.

For example, if an argument form involves five distinct atomic formulas (say, P, Q, R, S, T), then the associated truth table contains 32 rows. Indeed, every additional atomic formula doubles the size of the associated truth-table. This makes the truth-table method impractical in many cases, unless one has access to a computer. Even then, due to the "doubling" phenomenon, there are argument forms that even a very fast main-frame computer cannot solve, at least in a reasonable amount of time (say, less than 100 years!)

Another shortcoming of the truth-table method is that it does not require much in the way of reasoning. It is simply a matter of mechanically following a simple set of directions. Accordingly, this method does not afford much practice in reasoning, either formal or informal.

For these two reasons, we now examine a second technique for demonstrating the validity of arguments – the method of formal derivation, or simply derivation. Not only is this method less tedious and mechanical than the method of truth tables, it also provides practice in symbolic reasoning.

Skill in symbolic reasoning can in turn be transferred to skill in practical reasoning, although the transfer is not direct. By analogy, skill in any game of strategy (say, chess) can be transferred indirectly to skill in general strategy (such as war, political or corporate). Of course, chess does not apply directly to any real strategic situation.

Constructing a derivation requires more thinking than filling out truth-tables. Indeed, in some instances, constructing a derivation demands considerable ingenuity, just like a good combination in chess.

Unfortunately, the method of formal derivation has its own shortcoming: unlike truth-tables, which can show both validity and invalidity, derivations can only show validity. If one succeeds in constructing a derivation, then one knows that the corresponding argument is valid. However, if one fails to construct a derivation, it does not mean that the argument is invalid. In the past, humans repeatedly failed to fly; this did not mean that flight was impossible. On the other hand, humans have repeatedly tried to construct perpetual motion machines, and they have failed. Sometimes failure is due to lack of cleverness; sometimes failure is due to the impossibility of the task!
2. THE BASIC IDEA

Underlying the method of formal derivations is the following fundamental idea.

Granting the validity of a few selected argument forms, we can demonstrate the validity of other argument forms.

A simple illustration of this procedure might be useful. In an earlier chapter, we used the method of truth-tables to demonstrate the validity of numerous arguments. Among these, a few stand out for special mention. The first, and simplest one perhaps, is the following.

\[(\text{MP}) \quad P \quad \rightarrow \quad Q

\begin{array}{c}
P \\
Q
\end{array}
\]

This argument form is traditionally called *modus ponens*, which is short for *modus ponendo ponens*, which is a Latin expression meaning the mode of affirming by affirming. It is so called because, in this mode of reasoning, one goes from an affirmative premise to an affirmative conclusion.

It is easy to show that (MP) is a valid argument, using truth-tables. But we can use it to show other argument forms are also valid. Let us consider a simple example.

\[(a1) \quad P \quad \rightarrow \quad Q

\begin{array}{c}
P \\
Q \quad \rightarrow \quad R
\end{array}

\begin{array}{c}
R
\end{array}
\]

We can, of course, use truth-tables to show that (a1) is valid. Since there are three atomic formulas, 8 cases must be considered. However, we can also convince ourselves that (a1) is valid by reasoning as follows.

Proof: Suppose the premises are all true. Then, in particular, the first two premises are both true. But if P and P→Q are both true, then Q must be true. Why? Because Q follows from P and P→Q by *modus ponens*. So now we know that the following formulas are all true: P, P→Q, Q, Q→R. This means that, in particular, both Q and Q→R are true. But R follows from Q and Q→R, by *modus ponens*, so R (the conclusion) must also be true. Thus, if the premises are all true, then so is the conclusion. In other words, the argument form is valid.

What we have done is show that (a1) is valid assuming that (MP) is valid.

Another important classical argument form is the following.
(MT)  
\[ P \rightarrow Q \]
\[ \sim Q \]
\[ \sim P \]

This argument form is traditionally called *modus tollens*, which is short for *modus tollendo tollens*, which is a Latin expression meaning the mode of denying by denying. It is so called because, in this mode of reasoning, one goes from a negative premise to a negative conclusion.

Granting (MT), we can show that the following argument form is also valid.

(a2)  
\[ P \rightarrow Q \]
\[ Q \rightarrow R \]
\[ \sim R \]
\[ \sim P \]

Once again, we can construct a truth-table for (a2), which involves 8 lines. But we can also demonstrate its validity by the following reasoning.

Proof: Suppose that the premises are all true. Then, in particular, the last two premises are both true. But if \( Q \rightarrow R \) and \( \sim R \) are both true, then \( \sim Q \) is also true. For \( \sim Q \) follows from \( Q \rightarrow R \) and \( \sim R \), in virtue of *modus tollens*. So, if the premises are all true, then so is \( \sim Q \). That means that all the following formulas are true – \( P \rightarrow Q \), \( Q \rightarrow R \), \( \sim R \), \( \sim Q \). So, in particular, \( P \rightarrow Q \) and \( \sim Q \) are both true. But if these are true, then so is \( \sim P \) (the conclusion), because \( \sim P \) follows from \( P \rightarrow Q \) and \( \sim Q \), in virtue of *modus tollens*. Thus, if the premises are all true, then so is the conclusion. In other words, the argument form is valid.

Finally, let us consider an example of reasoning that appeals to both *modus ponens* and *modus tollens*.

(a3)  
\[ \sim P \]
\[ \sim P \rightarrow \sim R \]
\[ Q \rightarrow R \]
\[ \sim Q \]

Proof: Suppose that the premises are all true. Then, in particular, the first two premises are both true. But if \( \sim P \) and \( \sim P \rightarrow \sim R \) are both true, then so is \( \sim R \), in virtue of *modus ponens*. Then \( \sim R \) and \( Q \rightarrow R \) are both true, but then \( \sim Q \) is true, in virtue of *modus tollens*. Thus, if the premises are all true, then the conclusion is also true, which is to say the argument is valid.
3. ARGUMENT FORMS AND SUBSTITUTION INSTANCES

In the previous section, the alert reader probably noticed a slight discrepancy between the official argument forms (MP) and (MT), on the one hand, and the actual argument forms appearing in the proofs of the validity of (a1)-(a3).

For example, in the proof of (a3), I said that \(~R\) follows from \(~P\) and \(~P → ~R\), in virtue of *modus ponens*. Yet the argument forms are quite different.

\[
\text{(MP)} \quad \begin{align*}
\text{P} & \to \text{Q} \\
\text{P} & \\
\hline 
\text{Q}
\end{align*}
\]

\[
\text{(MP\*)} \quad \begin{align*}
\text{~P} & \to \text{~R} \\
\text{~P} & \\
\hline 
\text{~R}
\end{align*}
\]

(MP*) looks somewhat like (MP); if we squinted hard enough, we might say they looked the same. But, clearly, (MP*) is not exactly the same as (MP). In particular, (MP) has no occurrences of negation, whereas (MP*) has 4 occurrences. So, in what sense can I say that (MP*) is valid *in virtue of* (MP)?

The intuitive idea is that "the overall form" of (MP*) is the same as (MP). (MP*) is an argument form with the following overall form.

\[
\begin{align*}
\text{conditional formula} & \quad () \to [] \\
\text{antecedent} & \quad () \\
\hline 
\text{consequent} & \quad []
\end{align*}
\]

The fairly imprecise notion of overall form can be made more precise by appealing to the notion of a substitution instance. We have already discussed this notion earlier. The slight complication here is that, rather than substituting a concrete argument for an argument form, we substitute one argument form for another argument form,

The following is the official definition.

**Definition:**

If \(A\) is an argument form of sentential logic, then a *substitution instance* of \(A\) is any argument form \(A^*\) that is obtained from \(A\) by substituting formulas for letters in \(A\).

There is an affiliated definition for formulas.
Definition:
If \( F \) is a formula of sentential logic, then a **substitution instance** of \( F \) is any formula \( F' \) obtained from \( F \) by substituting formulas for letters in \( F \).

Note carefully: it is understood here that if a formula replaces a given letter in one place, then the formula replaces the letter in every place. One cannot substitute different formulas for the same letter. However, one is permitted to replace two different letters by the same formula. This gives rise to the notion of uniform substitution instance.

Definition:
A substitution instance is a **uniform substitution instance** if and only if distinct letters are replaced by distinct formulas.

These definitions are best understood in terms of specific examples. First, \((MP^*)\) is a (uniform) substitution of \((MP)\), obtained by substituting \(\neg P\) for \(P\), and \(\neg R\) for \(Q\). The following are examples of substitution instances of \((MP)\):

\[
\begin{array}{cccc}
\neg P \to \neg Q & (P \& Q) \to \neg R & (P \to Q) \to (P \to R) \\
\neg P & P \& Q & P \to Q \\
\neg Q & \neg R & P \to R
\end{array}
\]

Whereas \((MP^*)\) is a substitution instance of \((MP)\), the converse is not true: \((MP)\) is not a substitution instance of \((MP^*)\). There is no way to substitute formulas for letters in \((MP^*)\) in such a way that \((MP)\) is the result. \((MP^*)\) has four negations, and \((MP)\) has none. A substitution instance \(F^*\) always has at least as many occurrences of a connective as the original form \(F\).

The following are substitution instances of \((MP^*)\):

\[
\begin{array}{cccc}
\neg(P \& Q) \to \neg(P \to Q) & \neg\neg P \to \neg(Q \lor R) \\
\neg(P \& Q) & \neg P \\
\neg(P \to Q) & \neg(Q \lor R)
\end{array}
\]

Interestingly enough these are also substitution instances of \((MP)\). Indeed, we have the following general theorem.

**Theorem:**
If argument form \(A^*\) is a substitution instance of \(A\), and argument form \(A^{**}\) is a substitution instance of \(A^*\), then \(A^{**}\) is a substitution instance of \(A\).

With the notion of substitution instance in hand, we are now in a position to solve the original problem. To say that argument form \((MP^*)\) is valid *in virtue of* modus ponens \((MP)\) is not to say that \((MP^*)\) is identical to \((MP)\); rather, it is to
say that (MP*) is a substitution instance of (MP). The remaining question is whether the validity of (MP) ensures the validity of its substitution instances. This is answered by the following theorem.

**Theorem:**
If argument form \( A \) is valid, then every substitution instance of \( A \) is also valid.

The rigorous proof of this theorem is beyond the scope of introductory logic.

### 4. SIMPLE INFERENCE RULES

In the present section, we lay down the groundwork for constructing our system of formal derivation, which we will call system SL (short for ‘sentential logic’). At the heart of any derivation system is a set of *inference rules*. Each inference rule corresponds to a valid argument of sentential logic, although not every valid argument yields a corresponding inference rule. We select a subset of valid arguments to serve as inference rules.

But how do we make the selection? On the one hand, we want to be parsimonious. We want to employ as few inference rules as possible and still be able to generate all the valid argument forms. On the other hand, we want each inference rule to be simple, easy to remember, and intuitively obvious. These two desiderata actually push in opposite directions; the most parsimonious system is not the most intuitively clear; the most intuitively clear system is not the most parsimonious. Our particular choice will accordingly be a compromise solution.

We have to select from the infinitely-many valid argument forms of sentential logic a handful of very fertile ones, ones that will generate the rest. To a certain extent, the choice is arbitrary. It is very much like inventing a game – we get to make up the rules. On the other hand, the rules are not entirely arbitrary, because each rule must correspond to a valid argument form. Also, note that, even though we can choose the rules initially, once we have chosen, we must adhere to the ones we have chosen.

Every inference rule corresponds to a valid argument form of sentential logic. Note, however, that in granting the validity of an argument form (say, *modus ponens*), we mean to grant that specific argument form as well as *every* substitution instance.

In order to convey that each inference rule subsumes infinitely many argument forms, we will use an alternate font to formulate the inference rules; in particular, capital script letters (\( \mathcal{A}, \mathcal{B}, \mathcal{C} \), etc.) will stand for arbitrary formulas of sentential logic.

Thus, for example, the rule of *modus ponens* will be written as follows, where \( \mathcal{A} \) and \( \mathcal{C} \) are arbitrary formulas of sentential logic.
Given that the script letters ‘\( \mathcal{A} \)’ and ‘\( C \)’ stand for arbitrary formulas, (MP) stands for infinitely many argument forms, all looking like the following.

\[
\begin{array}{l}
\mathcal{A} \rightarrow C \\
\mathcal{A} \\
\hline
C
\end{array}
\]

Along the same lines, the rule *modus tollens* may be written as follows.

\[
\begin{array}{l}
\mathcal{A} \rightarrow C \\
\sim C \\
\hline
\sim \mathcal{A}
\end{array}
\]

Note: By ‘literal negation of formula \( \mathcal{A} \)’ is meant the formula that results from prefixing the formula \( \mathcal{A} \) with a tilde. The literal negation of a formula always has exactly one more symbol than the formula itself.

In addition to (MP) and (MT), there are two other similar rules that we are going to adopt, given as follows.

\[
\begin{array}{l}
\mathcal{A} \lor B \\
\sim \mathcal{A} \\
\hline
B
\end{array}
\]  
\[
\begin{array}{l}
\mathcal{A} \lor B \\
\sim B \\
\hline
\mathcal{A}
\end{array}
\]

This mode of reasoning is traditionally called *modus tollendo ponens*, which means the mode of affirming by denying. In each case, an affirmative conclusion is reached on the basis of a negative premise. The reader should verify, using truth-tables, that the simplest instances of these inference rules are in fact valid. The reader should also verify the intuitive validity of these forms of reasoning. MTP corresponds to the "process of elimination": one has a choice between two things, one eliminates one choice, leaving the other.

Before putting these four rules to work, it is important to point out two classes of errors that a student is liable to make.
• Errors of the First Kind

The four rules given above are to be carefully distinguished from argument forms that look similar but are clearly invalid. The following arguments are not instances of any of the above rules; worse, they are invalid.

<table>
<thead>
<tr>
<th>Invalid!</th>
<th>Invalid!</th>
<th>Invalid!</th>
<th>Invalid!</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P \rightarrow Q$</td>
<td>$P \rightarrow Q$</td>
<td>$P \lor Q$</td>
<td>$P \lor Q$</td>
</tr>
<tr>
<td>$Q$</td>
<td>$\neg P$</td>
<td>$P$</td>
<td>$Q$</td>
</tr>
<tr>
<td>$\underline{\neg Q}$</td>
<td>$\underline{\neg Q}$</td>
<td>$\underline{\neg Q}$</td>
<td>$\underline{\neg Q}$</td>
</tr>
</tbody>
</table>

These modes of inference are collectively known as *modus morons*, which means the mode of reasoning like a moron. It is easy to show that every one of them is invalid. You can use truth-tables, or you can construct counter-examples; either way, they are invalid.

• Errors of the Second Kind

Many valid arguments are not substitution instances of inference rules. This isn't too surprising. Some arguments, however, *look like* (but are not) substitution instances of inference rules. The following are examples.

<table>
<thead>
<tr>
<th>Valid but not MT!</th>
<th>Valid but not MT!</th>
<th>Valid but not MTP!</th>
<th>Valid but not MTP!</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\neg P \rightarrow Q$</td>
<td>$P \rightarrow \neg Q$</td>
<td>$\neg P \lor \neg Q$</td>
<td>$P \lor \neg Q$</td>
</tr>
<tr>
<td>$\neg Q$</td>
<td>$Q$</td>
<td>$\neg P$</td>
<td>$Q$</td>
</tr>
<tr>
<td>$\underline{P}$</td>
<td>$\underline{\neg Q}$</td>
<td>$\underline{\neg P}$</td>
<td>$\underline{\neg P}$</td>
</tr>
</tbody>
</table>

The following are corresponding correct applications of the rules.

<table>
<thead>
<tr>
<th>MT</th>
<th>MT</th>
<th>MTP</th>
<th>MTP</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\neg P \rightarrow Q$</td>
<td>$P \rightarrow \neg Q$</td>
<td>$\neg P \lor \neg Q$</td>
<td>$P \lor \neg Q$</td>
</tr>
<tr>
<td>$\neg Q$</td>
<td>$\neg Q$</td>
<td>$\neg P$</td>
<td>$\neg P$</td>
</tr>
<tr>
<td>$\underline{\neg P}$</td>
<td>$\underline{\neg Q}$</td>
<td>$\underline{\neg P}$</td>
<td>$\underline{\neg P}$</td>
</tr>
</tbody>
</table>

The natural question is, “aren’t $\neg \neg P$ and $P$ the same?” In asking this question, one might be thinking of arithmetic: for example, $\neg \neg 2$ and 2 are one and same *number*. But the corresponding *numerals* are not identical: the linguistic expression ‘$\neg \neg 2$’ is not identical to the linguistic expression ‘2’. Similarly, the Roman *numeral* ‘VII’ is not identical to the Arabic *numeral* ‘7’ even though both *numerals* denote the same *number*. Just like people, numbers have names; the names of numbers are numerals. We don’t confuse people and their names. We shouldn’t confuse numbers and their names (numerals).

Thus, the answer is that the formulas $\neg \neg P$ and $P$ are not the same; they are as different as the Roman numeral ‘VII’ and the Arabic numeral ‘7’.
Another possible reason to think $\sim \sim P$ and $P$ are the same is that they are logically equivalent, which may be shown using truth tables. This means they have the same truth-value no matter what. They have the same truth-value; does that mean they are the same? Of course not! That is like arguing from the premise that John and Mary are legally equivalent (meaning that they are equal under the law) to the conclusion that John and Mary are the same. Logical equivalence, like legal equivalence, is not identity.

Consider a very similar question whose answer revolves around the distinction between equality and identity: are four quarters and a dollar bill the same? The answer is, “yes and no”. Four quarters are monetarily equal to a dollar bill, but they are definitely not identical. Quarters are made of metal, dollar bills are made of paper; they are physically quite different. For some purposes they are interchangeable; that does not mean they are the same.

The same can be said about $\sim \sim P$ and $P$. They have the same value (in the sense of truth-value), but they are definitely not identical. One has three symbols, the other only one, so they are not identical. More importantly, for our purposes, they have different forms – one is a negation; the other is atomic.

A derivation system in general, and inference rules in particular, pertain exclusively to the forms of the formulas involved.

In this respect, derivation systems are similar to coin-operated machines – vending machines, pay phones, parking meters, automatic toll booths, etc. A vending machine, for example, does not "care" what the value of a coin is. It only "cares" about the coin's form; it responds exclusively to the shape and weight of the coin. A penny worth one dollar to collectors won't buy a soft drink from a vending machine. Similarly, if the machine does not accept pennies, it is no use to put in 25 of them, even though 25 pennies have the same monetary value as a quarter. Similarly frustrating at times, a dollar bill is worthless when dealing with many coin-operated machines.

A derivation system is equally "stubborn"; it is blind to content, and responds exclusively to form. The fact that truth-tables tell us that $P$ and $\sim \sim P$ are logically equivalent is irrelevant. If $P$ is required by an inference-rule, then $\sim \sim P$ won't work, and if $\sim \sim P$ is required, then $P$ won't work, just like 25 pennies won't buy a stick of gum from a vending machine. What one must do is first trade $P$ for $\sim \sim P$. We will have such conversion rules available.
5. SIMPLE DERIVATIONS

We now have four inference rules, MP, MT, MTP1, and MTP2. How do we utilize these in demonstrating other arguments of sentential logic are also valid? In order to prove (show, demonstrate) that an argument is valid, one derives its conclusion from its premises. We have already seen intuitive examples in an earlier section. We now redo these examples formally.

The first technique of derivation that we examine is called simple derivation. It is temporary, and will be replaced in the next section. However, it demonstrates the key intuitions about derivations.

Simple derivations are defined as follows.

**Definition:**

A simple derivation of conclusion \( C \) from premises \( P_1, P_2, ..., P_n \) is a list of formulas (also called lines) satisfying the following conditions.

1. the last line is \( C \);
2. every line (formula) is either:
   - a premise (one of \( P_1, P_2, ..., P_n \)),
   - or:
     follows from previous lines according to an inference rule.

The basic idea is that in order to prove that an argument is valid, it is sufficient to construct a simple derivation of its conclusion from its premises. Rather than dwell on abstract matters of definition, it is better to deal with some examples by way of explaining the method of simple derivation.

**Example 1**

Argument: \( P ; P \rightarrow Q ; Q \rightarrow R / R \)

Simple Derivation:

<table>
<thead>
<tr>
<th>Line</th>
<th>Formula</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( P )</td>
<td>Pr</td>
</tr>
<tr>
<td>2</td>
<td>( P \rightarrow Q )</td>
<td>Pr</td>
</tr>
<tr>
<td>3</td>
<td>( Q \rightarrow R )</td>
<td>Pr</td>
</tr>
<tr>
<td>4</td>
<td>( Q )</td>
<td>1,2,MP</td>
</tr>
<tr>
<td>5</td>
<td>( R )</td>
<td>3,4,MP</td>
</tr>
</tbody>
</table>

This is an example of a simple derivation. The last line is the conclusion; every line is either a premise or follows by a rule. The annotation to the right of each formula indicates the precise justification for the presence of the formula in the
derivation. There are two possible justifications at the moment; the formula is a premise (annotation: ‘Pr’); the formula follows from previous formulas by a rule (annotation: line numbers, rule).

**Example 2**

Argument: $P \rightarrow Q ; Q \rightarrow R ; \sim R / \sim P$

Simple Derivation:

1. $P \rightarrow Q$ \hspace{1cm} Pr
2. $Q \rightarrow R$ \hspace{1cm} Pr
3. $\sim R$ \hspace{1cm} Pr
4. $\sim Q$ \hspace{1cm} 2,3,MT
5. $\sim P$ \hspace{1cm} 1,4,MT

**Example 3**

Argument: $\sim P ; \sim P \rightarrow \sim R ; Q \rightarrow R / \sim Q$

Simple Derivation:

1. $\sim P$ \hspace{1cm} Pr
2. $\sim P \rightarrow \sim R$ \hspace{1cm} Pr
3. $Q \rightarrow R$ \hspace{1cm} Pr
4. $\sim R$ \hspace{1cm} 1,2,MP
5. $\sim Q$ \hspace{1cm} 3,4,MT

These three examples take care of the examples from Section 2. The following one is more unusual.

**Example 4**

Argument: $(P \rightarrow Q) \rightarrow P ; P \rightarrow Q / Q$

Simple Derivation:

1. $(P \rightarrow Q) \rightarrow P$ \hspace{1cm} Pr
2. $P \rightarrow Q$ \hspace{1cm} Pr
3. $P$ \hspace{1cm} 1,2,MP
4. $Q$ \hspace{1cm} 2,3,MP

What is unusual about this one is that line (2) is used twice, in connection with MP, once as minor premise, once as major premise. One can appeal to the same line over and over again, if the need arises.

We conclude this section with examples of slightly longer simple derivations.
Example 5

Argument: $P \rightarrow (Q \lor R) ; P \rightarrow \neg R ; P \lor Q$

Simple Derivation:

1. $P \rightarrow (Q \lor R)$  Pr
2. $P \rightarrow \neg R$  Pr
3. $P$  Pr
4. $\neg R$  2,3,MP
5. $Q \lor R$  1,3,MP
6. $Q$  4,5,MTP2

Example 6

Argument: $\neg P \rightarrow (Q \lor R) ; P \rightarrow Q ; \neg Q \lor R$

Simple Derivation:

1. $\neg P \rightarrow (Q \lor R)$  Pr
2. $P \rightarrow Q$  Pr
3. $\neg Q$  Pr
4. $\neg P$  2,3,MT
5. $Q \lor R$  1,4,MP
6. $R$  3,5,MTP1

Example 7

Argument: $(P \lor R) \lor (P \rightarrow Q) ; \neg(P \rightarrow Q) ; R \rightarrow (P \rightarrow Q)$ / $P$

Simple Derivation:

1. $(P \lor R) \lor (P \rightarrow Q)$  Pr
2. $\neg(P \rightarrow Q)$  Pr
3. $R \rightarrow (P \rightarrow Q)$  Pr
4. $P \lor R$  1,2,MTP2
5. $\neg R$  2,3,MT
6. $P$  4,5,MTP2

Example 8

Argument: $P \rightarrow \neg Q ; \neg Q \rightarrow (R \& S) ; \neg(R \& S) ; P \lor T$ / $T$

Simple Derivation:

1. $P \rightarrow \neg Q$  Pr
2. $\neg Q \rightarrow (R \& S)$  Pr
3. $\neg(R \& S)$  Pr
4. $P \lor T$  Pr
5. $\neg \neg Q$  2,3,MT
6. $\neg P$  1,5,MT
7. $T$  4,6,MTP1