Method of Reductio Ad Absurdum

Consider this lengthy argument:

P1: If taxes are cut then the government’s revenue is reduced.
P2: If the government’s revenue is reduced then the deficit will grow.
P3: The government’s revenue was not reduced.
C: Taxes were raised and the deficit was reduced.

We let p: “Taxes are cut”, q: “Government’s revenue was reduced” and r: “The deficit will grow”. With these basic propositions we can rewrite the argument in a more compact form as follows:

\[ P1: p \rightarrow q \]
\[ P2: q \rightarrow r \]
\[ P3: \sim q \]
\[ C: \sim p \land \sim r \]

We next form this argument into one long single statement, keeping in mind that (a) all premises are connected by ‘and’ and (b) the premises are conclusion are connected by an ‘If-then’. We get:

\[ \text{If } [(p \rightarrow q) \land (q \rightarrow r) \land (\sim q)] \text{ then } [(\sim p) \land (\sim r)] \]

Which can be compacted into this tighter form:

“If [P1 and P2 and P3] then [C].”

Symbolically, the argument is:

\[ [(p \rightarrow q) \land (q \rightarrow r) \land (\sim q)] \rightarrow (\sim p \land \sim r) \]

Rather than do a lengthy truth table, we employ a method of reasoning called *reductio ad absurdum*. In this method, we “force” the whole argument to be false, trying to make it invalid. We then work through each part of the argument and see what happens. If we are successful in forcing the argument to be false, then the argument is invalid. If we come to a contradiction, we conclude that it’s impossible to force the argument to be false, and therefore it must be true always, hence valid.

Reductio ad absurdum is a bit more free-form than a truth table, but with practice it can be a very powerful tool for analyzing complex arguments.

Let’s do this step-by-step.

1. First, look at the whole argument. Since we want it to be false, we see that an ‘if-then’ can only be false when the first part (the premises) are true and when the second part (the conclusion) is false. We are working outside-in.

   If \([(p \rightarrow q) \land (q \rightarrow r) \land (\sim q)] \text{ then } [(\sim p) \land (\sim r)] = F \]

   This means that the premises “P1 and P2 and P3” are T and the conclusion “C” is F

2. Since the block of premises are true, and since they are connected by ‘and’, this can only happen when all individual premises are true. Therefore we have the following:

   “if p then q” = T,  “if q then r” = T,  “not q” = T
3. We now look at each premise individually and attempt to deduce the value of \( p, q \) or \( r \). In this case, the third premise “not \( q \)” allows us to determine \( q \). Since we know “not \( q \) = T”, then \( q \) itself must be false.

4. We now use this fact to see if we can lock in the truth values for \( p \) or \( r \). Look at the first premise, P1: “If \( p \) then \( q \)”. Since we know this premise is true (see step 2), and we know \( q \) is false (step 3), then we have “If \(<\text{blank}>\) then \( F\)” = T, which can only happen when \(<\text{blank}>\) is F. This forces \( p \) to be false. (“\text{blank}”, in this case, is \( p \))

5. The second premise still does not allow us to lock in \( r \), so we skip over to the conclusion, which we know is false (step 1). We know that \( p \) is false (step 4) so therefore, ‘not \( p \)’ is true. Remember, the conclusion is “(not \( p \) and (not \( r \))”. We know “not \( p \)” is true, so we have “T and \(<\text{blank}>\)” = F. This forces \(<\text{blank}>\) to be F, meaning that “not \( r \)” is false. Hence, we have now locked in \( r \) to be true.

6. We still have to check the second premise. It reads “If \( q \) then \( r \)” . We know that this is true (step 2). We know that \( q \) is false (step 3). We know that \( r \) is true (step 5). Does “If \( F \) then \( T \) = T”? It does. There is no contradiction. We have successfully shown that it is possible to force the argument to be false. Hence, the argument is invalid.

Here is the truth table:

<table>
<thead>
<tr>
<th>( p )</th>
<th>( q )</th>
<th>( r )</th>
<th>If ( p ) then ( q ) (P1)</th>
<th>If ( q ) then ( r ) (P2)</th>
<th>not ( q ) (P3)</th>
<th>P1 and P2 and P3 (“Premises”)</th>
<th>not ( p )</th>
<th>not ( r )</th>
<th>(not ( p ) and (not ( r )) (“Conclusion”)</th>
<th>If [premises] then [conclusion]</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</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>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</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>
<td>T</td>
<td>T</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>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

As you can see, this argument is invalid; it has seven T’s and one F, and that F occurs when \( p = F, q = F \) and \( r = T \), just as we deduced previously.

Here’s an example of a valid argument going through the method of reduction ad absurdum:

\[
\text{If } [(\text{if } p \text{ then } q) \text{ and } (\text{if } q \text{ then } r) \text{ and } (\text{not } r)] \text{ then } [\text{not } p]
\]

The premises are: “if \( p \) then \( q \)” , “if \( q \) then \( r \)” and “not \( r \)” . The conclusion is “not \( p \)”.

Here’s the reductio ad absurdum argument: Assume this argument is false. Therefore, the premises are T and the conclusion is F. In this case, we can lock in \( p \) right now. Since “not \( p \)” is F, we have that \( p = T \). Since all premises are true, we have that the third premise “not \( r \)” is T, hence \( r = F \). In the first premise, since we know \( p = T \) and since we know “If \( T \) then \( <\text{blank}>\)” = T, we must have that \( q = T \). Now we need to check the second premise, “If \( q \) then \( r \)” . We know it’s true. And we also have that \( q \) is true while \( r \) is false. This is contradictory. This cannot happen. Hence, it is impossible to force this argument to be false. Therefore, the argument is true and hence, valid.

Your project: (20 points). Use this method to show whether the following argument is valid or not. Connect these premises and the conclusion into one long statement then try to ‘lock in’ the truth values of \( p, q \) and \( r \). Is there a contradiction?

P1: \( p \rightarrow q \)  
P2: \( q \rightarrow \sim r \)  
P3: \( \sim q \)  
C: \( \sim p \land r \)