PROBLEM 1

Given a set of formulas 
\[ G = \{(a \Rightarrow (a \cup b)), \ (a \cup b), \ \neg b, \ (c \Rightarrow b)\} \]

1. Show that \( G \) is CONSISTENT under classical semantics. Use shorthand notation.

Solution

We find a restricted model for \( G \).

The formula \((a \Rightarrow (a \cup b))\) is a tautology, hence any \( v \) is its model. We have that \( \neg b = T \) only if \( b = F \).

We have that \((a \cup b) = (a \cup F) = T\) only if \( a = T \).

Consequently, \((c \Rightarrow b) = (c \Rightarrow F) = T\) only if \( c = F \).

Hence, any \( v \), such that \( a = T, \ b = T, \ and \ c = F \) is a model for \( G \). This proves that \( G \) is CONSISTENT.

2. Find a formula \( A \) that is iINDEPENDENT of \( G \). Must prove it. Use shorthand notation.

Solution

This is my solution. This is not the only one!

Let \( A \) be any atomic formula \( d \in VAR - \{a, b, c\} \).

Any \( v \), such that \( a = T, \ b = T, \ and \ c = F, \ d = T \) is a model for \( G \cup \{A\} \).

Any \( v \), such that \( a = T, \ b = T, \ and \ c = F, \ d = F \) is a model for \( G \cup \{\neg A\} \). This proves that the formula \( d \in VAR - \{a, b, c\} \) is iINDEPENDENT of \( G \).

3. Find an infinite number of formulas that are iINDEPENDENT of \( G \). Justify your answer.

Solution

This is my solution. This is not the only one!

There is countably infinitely many atomic formulas \( A = d \) where \( d \in VAR - \{a, b, c\} \).
**PROBLEM 2**

Given a language \( \mathcal{L} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}} \). We define a **L₄ semantics** as follows.

Logical values are \( F, \bot₁, \bot₂, T \) and they are ordered: \( F < \bot₁ < \bot₂ < T \).

The **connectives** are defined as follows:

\[-\bot₁ = \bot₁, \quad -\bot₂ = \bot₂, \quad -F = T, \quad -T = F.\]

For any \( x, y \in \{F, \bot₁, \bot₂, T\} \), \( x \cap y = \min\{x, y\} \), \( x \cup y = \max\{x, y\} \), and

\[x \Rightarrow y = \begin{cases} 
- x \cup y \text{ if } x > y \\
T \text{ otherwise}
\end{cases}\]

1. **Write Truth Tables for implication and negation.**

   **Solution**

   \[
   \begin{array}{c|cccc}
   \Rightarrow & F & \bot₁ & \bot₂ & T \\
   \hline
   F & T & T & T & T \\
   \bot₁ & T & T & T & T \\
   \bot₂ & T & T & T & T \\
   T & F & \bot₁ & \bot₂ & T \\
   \end{array}
   \]

   \[
   \begin{array}{c|cccc}
   - & F & \bot₁ & \bot₂ & T \\
   \hline
   T & \bot₁ & \bot₂ & F & T \\
   \end{array}
   \]

2. **Prove/disprove:** \( \models_{L₄} ((a \Rightarrow b) \Rightarrow (\neg a \cup b)) \). Use **shorthand notation**.

   **Solution**

   Let \( \nu \) be a truth assignment such that \( \nu(a) = \nu(b) = \bot₁ \).

   We evaluate \( \nu'((a \Rightarrow b) \Rightarrow (\neg a \cup b)) = ((\bot₁ \Rightarrow \bot₁) \Rightarrow (\neg \bot₁ \cup \bot₁)) = (T \Rightarrow (\bot₁ \cup \bot₁)) = (T \Rightarrow \bot₁) = \bot₁ \).

   This proves that \( \nu \) is a **counter-model** for our formula and that \( \not\models_{L₄} ((a \Rightarrow b) \Rightarrow (\neg a \cup b)) \).

   Observe that there are other counter-models. For example, \( \nu \) such that \( \nu(a) = \nu(b) = \bot₂ \) is also a counter model, as \( \nu'((a \Rightarrow b) \Rightarrow (\neg a \cup b)) = ((\bot₂ \Rightarrow \bot₂) \Rightarrow (\neg \bot₂ \cup \bot₂)) = (T \Rightarrow (\bot₂ \cup \bot₂)) = (T \Rightarrow \bot₂) = \bot₂ \).

3. **Prove** that the equivalence defining \( \cup \) in terms of negation and implication in classical logic **does not hold** under \( L₄ \), i.e. prove that \( (A \cup B) \not\models_{L₄} (\neg A \Rightarrow B) \).

   **Solution**

   Any \( \nu \) such that \( \nu'(A) = \bot₂ \) and \( \nu'(B) = \bot₁ \) is a **counter-model**. This is not the only counter-model.
PROBLEM 3

Consider the Hilbert system $H_1 = (\mathcal{L}_{[\Rightarrow]}, \mathcal{F}, \{A1, A2\}, (MP) \frac{A : (A=B)}{B})$ where for any $A, B \in \mathcal{F}$

A1: $(A \Rightarrow (B \Rightarrow A))$. A2: $((A \Rightarrow (B \Rightarrow C)) \Rightarrow ((A \Rightarrow B) \Rightarrow (A \Rightarrow C)))$.

1. We have proved that the Deduction Theorem holds for $H_1$.

Use Deduction Theorem to prove $(A \Rightarrow (C \Rightarrow B)) \vdash_H (C \Rightarrow (A \Rightarrow B))$.

Solution

We apply the Deduction Theorem twice, i.e. we get

$(A \Rightarrow (C \Rightarrow B)) \vdash_H (C \Rightarrow (A \Rightarrow B))$ if and only if

$(A \Rightarrow (C \Rightarrow B)), C \vdash_H (A \Rightarrow B)$ if and only if

$(A \Rightarrow (C \Rightarrow B)), C, A \vdash_B B$

We now construct a proof of $(A \Rightarrow (C \Rightarrow B)), C, A \vdash_B B$ as follows

$B_1: (A \Rightarrow (C \Rightarrow B))$ hypothesis

$B_2: C$ hypothesis

$B_3: A$ hypothesis

$B_4: (C \Rightarrow B) \quad B_1, B_3$ and (MP)

$B_5: C \quad B_2, B_4$ and (MP)

2. Explain why 1. proves that $(\neg a \Rightarrow ((b \Rightarrow \neg a) \Rightarrow b)) \vdash_H ((b \Rightarrow \neg a) \Rightarrow (\neg a \Rightarrow b))$.

Solution

This is 1. for $A = \neg a, C = (b \Rightarrow \neg a)$, and $B = b$.

3. $H_1$ is sound under classical semantics. Explain why $H_1$ is not complete.

Solution

The system $S$ is not complete under classical semantics means that not all classical tautologies have a proof in $S$.

We have proved that one needs negation and one of other connectives $\cup, \cap, \Rightarrow$ to express all classical connectives, and hence all classical tautologies.

Our language contains only implication and one can’t express negation in terms of implication alone and hence we can’t provide a proof of any tautology i.e. its logically equivalent form in our language $\mathcal{L}_{[\Rightarrow]}$. 
4. Let \( H2 \) be the proof system obtained from the system \( H1 \) by **extending the language** to contain the negation \( \neg \) and **adding** one additional axiom:

\[
A3 \ ( (\neg B \Rightarrow \neg A) \Rightarrow (\neg B \Rightarrow A) \Rightarrow B )
\]

**Explain** shortly why **Deduction Theorem** holds for \( H2 \).

**Solution**

The proof of the Deduction Theorem for \( H1 \) used only axioms \( A1, A2 \).

Adding \( \neg \) to the language and adding axiom \( A3 \) does not change anything in the proof.

Hence **Deduction Theorem** holds for \( H2 \).

5. We know that \( H2 \) is **complete**.

Let \( H3 \) be the proof system obtained from the system \( H2 \) **adding** additional axiom

\[
A4 \ ( \neg (A \Rightarrow B) \Rightarrow \neg (A \Rightarrow \neg B) )
\]

Does **Deduction Theorem** holds for \( H3 \)? Justify.

**Solution**

Yes, it **does**, by the same argument as for \( H2 \).

Does **Completeness Theorem** holds for \( H3 \)? Justify.

**Solution**

No, it **doesn’t**. The system \( H3 \) is **not sound**. Axiom \( A4 \) is not a tautology.

Any \( v \) such that \( A=T \) and \( B=F \) is a **counter model** for \( (\neg(A \Rightarrow B) \Rightarrow \neg(A \Rightarrow \neg B) ) \).

**PROBLEM 4**

Let \( GL \) be the Gentzen style proof system for classical logic defined in chapter 6.

Prove, by constructing a proper decomposition tree that

\[
\vdash_{GL} ((\neg a \Rightarrow \neg \neg b) \Rightarrow (\neg b \Rightarrow a))
\]

**Solution**

By definition we have that

\[
\vdash_{GL} ((\neg a \Rightarrow \neg \neg b) \Rightarrow (\neg b \Rightarrow a)) \quad \text{if and only if} \quad \vdash_{GL} \rightarrow ((\neg a \Rightarrow \neg \neg b) \Rightarrow (\neg b \Rightarrow a)).
\]
We construct the decomposition tree as follows

\[
T \rightarrow A
\]

\[
\rightarrow ((\neg a \Rightarrow \neg b) \Rightarrow (\neg b \Rightarrow a))
\]

\[
\mid (\rightarrow \Rightarrow)
\]

\[
(\neg a \Rightarrow \neg b) \rightarrow (\neg b \Rightarrow a)
\]

\[
\mid (\rightarrow \Rightarrow)
\]

\[
\neg b, (\neg a \Rightarrow \neg b) \rightarrow a
\]

\[
\mid (\rightarrow \neg)
\]

\[
(\neg a \Rightarrow \neg b) \rightarrow b, a
\]

\[\land (\Rightarrow \rightarrow)\]

\[
\rightarrow \neg a, b, a
\]

\[
\mid (\rightarrow \neg)
\]

\[
a \rightarrow b, a
\]

\[
\text{axiom}
\]

\[
\neg b \rightarrow b, a
\]

\[
\mid (\neg \rightarrow)
\]

\[
(\neg b, b, a
\]

\[
\mid (\rightarrow \neg)
\]

\[
b \rightarrow b, a
\]

\[
\text{axiom}
\]

All leaves of the tree are axioms, hence we have found the proof of \(A\) in \(GL\).

**PROBLEM 5**

Prove, by constructing proper decomposition trees that

\[
\Rightarrow_{GL} ((b \Rightarrow a) \Rightarrow (\neg a \Rightarrow b))
\]

**Solution**

Observe that for any formula \(A\), its decomposition trees \(T_{\rightarrow A}\) in \(GL\) is not unique.

Hence when constructing decomposition trees we have to cover all possible cases.

We construct the decomposition tree \(T_1\) for

\[
\rightarrow ((b \Rightarrow a) \Rightarrow (\neg a \Rightarrow b))
\]

as follows.
The tree contains a non-axiom leaf $\rightarrow b,a,b$, hence it is not a proof in GL.

We have only one more tree to construct. Here it is.

$$T_2$$

$$\rightarrow ((b \Rightarrow a) \Rightarrow (\neg a \Rightarrow b))$$

$$| (\rightarrow \Rightarrow)$$

(one choice)

$$(b \Rightarrow a) \rightarrow (\neg a \Rightarrow b)$$

$$| (\rightarrow \Rightarrow)$$

(first of two choices)

$$\neg a, (b \Rightarrow a) \rightarrow b$$

$$| (\neg \rightarrow)$$

(one choice)

$$(b \Rightarrow a) \rightarrow a,b$$

$$\wedge (\Rightarrow \rightarrow)$$

(second of two choices)

$$\rightarrow b,a,b \quad a \rightarrow a,b$$

non-axiom axiom
\[ \rightarrow (\neg a \Rightarrow b), b \quad a \rightarrow (\neg a \Rightarrow b) \]

\[ (\rightarrow \Rightarrow) \quad | (\rightarrow \Rightarrow) \quad (one \ choice) \quad (one \ choice) \]

\[ \neg a \rightarrow b, b \quad a, \neg a \rightarrow b \]

\[ | (\neg \rightarrow) \quad | (\neg \rightarrow) \quad (one \ choice) \quad (one \ choice) \]

\[ \rightarrow a, b, b \quad a \rightarrow a, b \]

\[ non - axiom \quad axiom \]

All possible trees end with a non-axiom leave which proves that

\[ \not \vDash_{GL} ((b \Rightarrow a) \Rightarrow (\neg a \Rightarrow b)). \]

**PROBLEM 6**

Let \( GL \) be the Gentzen style proof system for classical logic defined in chapter 6.

Prove, by constructing a **counter-model** defined by a proper decomposition tree that

\[ \not \vDash ((a \Rightarrow (\neg b \land c)) \Rightarrow (\neg b \Rightarrow (a \lor \neg c))) \]

**Explain** why your counter-model construction is valid

**Solution**

\[ T_{\rightarrow A} \]

\[ \rightarrow ((a \Rightarrow (\neg b \land c)) \Rightarrow (\neg b \Rightarrow (a \lor \neg c))) \]

\[ | (\rightarrow \Rightarrow) \]

\[ (a \Rightarrow (\neg b \land c)) \rightarrow (\neg b \Rightarrow (a \lor \neg c)) \]

\[ | (\rightarrow \Rightarrow) \quad (one \ of \ two \ choices) \]

\[ \neg b, (a \Rightarrow (\neg b \land c)) \rightarrow (a \lor \neg c) \]

\[ | (\rightarrow \lor) \quad (one \ of \ two \ choices) \]

\[ \neg b, (a \Rightarrow (\neg b \land c)) \rightarrow a, \neg c \quad (one \ of \ two \ choices) \]

7
\[ (a \Rightarrow (\neg b \land c)) \rightarrow a, \neg c, b \]

one of two choices

\[ (\neg \rightarrow) \]

\[ c, (a \Rightarrow (\neg b \land c)) \rightarrow a, b \]

\[ \land (\Rightarrow \rightarrow) \]

\[ c \rightarrow a, a, b \]

non-axiom

\[ (\neg b \land c) \rightarrow a, b \]

\[ (\land \rightarrow) \]

\[ \neg b, c \rightarrow a, b \]

\[ (\neg \rightarrow) \]

\[ c \rightarrow b, a, b \]

non-axiom

The counter-model model determined by the non-axiom leaf \( c \rightarrow a, a, b \) is any truth assignment that evaluates it to \( F \).

Observe that (we use a shorthand notation) \( c \rightarrow a, a, b = F \) if and only if \( c = T \) and \( a = F \) and \( b = F \).

The counter-model model determined by the non-axiom leaf \( c \rightarrow b, a, b \) is any also any truth assignment that \( c = T \) and \( a = F \) and \( b = F \).

The counter-model construction is valid because of the strong soundness of GL.

PROBLEM 7

Let LI be the Gentzen system for intuitionistic logic as defined in chapter 7.

Determine whether

\[ \vdash_{LI} ((\neg a \land \neg c) \Rightarrow \neg (a \cup c)) \]

This means that you have to construct some, or all decomposition trees of

\[ (\neg a \land \neg c) \Rightarrow \neg (a \cup c) \]

If you find a decomposition tree such that all its leaves are axioms, you have a proof.

If all possible decomposition trees have a non-axiom leaf, the proof in LI does not exist.

Solution Consider the following decomposition tree of

\[ ((\neg a \land \neg c) \Rightarrow \neg (a \cup c)) \]
The tree $T_1$ has a non-axiom leaf, so it does not constitute a proof in $LI$. But this fact does not yet prove that proof doesn’t exist, as the decomposition tree in $LI$ is not always unique.

Let’s consider now the following tree.

$T_2$

$$\rightarrow ((\neg a \land \neg c) \Rightarrow (\neg a \cup c))$$

$$| (\rightarrow \Rightarrow)$$

$$| (\neg \rightarrow)$$

$$| (exch \rightarrow)$$

$$| (\cap \rightarrow)$$

$$| (\rightarrow \Rightarrow)$$

$$| (\neg \rightarrow)$$

$$| (\neg \rightarrow)$$

$$| (\cap \rightarrow)$$

$$a \rightarrow c$$

$$c \rightarrow c$$

non-axiom

axiom
\[\neg a, \neg c, (a \cup c) \rightarrow \]
| (exch $\rightarrow$)
\[\neg a, (a \cup c), \neg c \rightarrow \]
| (exch $\rightarrow$)
\[(a \cup c), \neg a, \neg c \rightarrow \]
\[\lor (\land \rightarrow)\]
\[a, \neg a, \neg c \rightarrow \quad c, \neg a, \neg c \rightarrow \]
| (exch $\rightarrow$) | (exch $\rightarrow$)
\[\neg a, a, \neg c \rightarrow \quad c, \neg c, \neg a \rightarrow \]
| (\rightarrow) | (exch $\rightarrow$)
\[a, \neg c \rightarrow a \quad \neg c, c, \neg a \rightarrow \]
\[\text{axiom} \quad \text{axiom} \]
\[c, \neg a \rightarrow c \]

All leaves of $T_2$ are axioms, what proves that $T_2$ is a proof of $A$ and hence we proved that

$\vdash_{\text{LI}} ((\neg a \cap \neg c) \Rightarrow (a \cup c))$

**Observe** that your FIRST tree is $T_2$, you have found the PROOF, so there is no need to examine any other trees