The proof system GL for the classical propositional logic is a version of the original Gentzen (1934) systems LK.

A constructive proof of the completeness theorem for the system GL is very similar to the proof of the completeness theorem for the system RS.

Expressions of the system like in the original Gentzen system LK are Gentzen sequents.

Hence we use also a name Gentzen sequent calculus.
Language of GL: $\mathcal{L} = \mathcal{L}_{\{\cup, \cap, \Rightarrow, \neg\}}$.

We add a new symbol to the alphabet: $\rightarrow$. It is called a Gentzen arrow.

The sequents are built out of finite sequences (empty included) of formulas, i.e. elements of $\mathcal{F}^*$, and the additional sign $\rightarrow$.

We denote, as in the RS system, the finite sequences of formulas by Greek capital letters $\Gamma, \Delta, \Sigma$, with indices if necessary.

Sequent definition: a sequent is the expression

$$\Gamma \rightarrow \Delta,$$

where $\Gamma, \Delta \in \mathcal{F}^*$. 
**Meaning of sequents**  Intuitively, we interpret a sequent

\[ A_1, \ldots, A_n \rightarrow B_1, \ldots, B_m, \]

where \( n, m \geq 1 \) as a formula

\[(A_1 \cap \ldots \cap A_n) \Rightarrow (B_1 \cup \ldots \cup B_m).\]

**The sequent:**  \( A_1, \ldots, A_n \rightarrow \) (where \( n \geq 1 \)) means *that* \( A_1 \cap \ldots \cap A_n \) *yields a contradiction.*

**The sequent**  \( \rightarrow B_1, \ldots, B_m \) (where \( m \geq 1 \)) means \( \models (B_1 \cup \ldots \cup B_m). \)

**The empty sequent:**  \( \rightarrow \) means *a contradiction.*
Given non empty sequences: \( \Gamma, \Delta \), we denote by

\[ \sigma_\Gamma \]

*any conjunction* of all formulas of \( \Gamma \), and by

\[ \delta_\Delta \]

*any disjunction* of all formulas of \( \Delta \).

The intuitive semantics (meaning, interpretation) of a sequent \( \Gamma \rightarrow \Delta \) (where \( \Gamma, \Delta \) are nonempty) is

\[ \Gamma \rightarrow \Delta \equiv (\sigma_\Gamma \Rightarrow \delta_\Delta). \]
Formal semantics for sequents (expressions of GL)

Let \( v : VAR \rightarrow \{ T, F \} \) be a truth assignment, \( v^* \) its (classical semantics) extension to the set of formulas \( \mathcal{F} \).

We extend \( v^* \) to the set

\[
SEQ = \{ \Gamma \rightarrow \Delta : \Gamma, \Delta \in \mathcal{F}^* \} \]

of all sequents as follows: for any sequent \( \Gamma \rightarrow \Delta \in SEQ \),

\[
v^*(\Gamma \rightarrow \Delta) = v^*(\sigma_{\Gamma}) \Rightarrow v^*(\delta_{\Delta}).
\]

In the case when \( \Gamma = \emptyset \) or \( \Delta = \emptyset \) we define:

\[
v^*(\rightarrow \Delta) = (T \Rightarrow v^*(\delta_{\Delta})),
\]

\[
v^*(\Gamma \rightarrow ) = (v^*(\sigma_{\Gamma}) \Rightarrow F).
\]
The sequent $\Gamma \rightarrow \Delta$ is *satisfiable* if there is a truth assignment $v : VAR \rightarrow \{T, F\}$ such that $v^*(\Gamma \rightarrow \Delta) = T$.

**Model** for $\Gamma \rightarrow \Delta$ is any $v$, such that

$$v^*(\Gamma \rightarrow \Delta) = T.$$

**We write** it

$$v \models \Gamma \rightarrow \Delta.$$

**Counter- model** is any $v$ such that

$$v^*(\Gamma \rightarrow \Delta) = F.$$

**We write** it

$$v \not\models \Gamma \rightarrow \Delta.$$
Tautology is any sequent $\Gamma \rightarrow \Delta$, such that $v^*(\Gamma \rightarrow \Delta) = T$ for all truth assignments $v : VAR \rightarrow \{T, F\}$, i.e.

$$| = \Gamma \rightarrow \Delta.$$ 

**Example** Let $\Gamma \rightarrow \Delta$ be a sequent

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

The truth assignment $v$ for which $v(a) = T$ and $v(b) = T$ is a model for $\Gamma \rightarrow \Delta$, as shows the following computation.

$$v^*(a, (b\cap a) \rightarrow \neg b, (b \Rightarrow a)) = v^*(\sigma_{\{a, (b\cap a)\}}) \Rightarrow$$

$$v^*(\delta_{\{-b, (b \Rightarrow a)\}}) = v(a) \cap (v(b) \cap v(a)) \Rightarrow$$

$$\neg v(b) \cup (v(b) \Rightarrow v(a)) = T \cap T\cap T \Rightarrow \neg T \cup (T \Rightarrow T) = T \Rightarrow (F \cup T) = T \Rightarrow T = T.$$
Observe that the only \( v \) for which \( v^*(\Gamma) = v^*(a, (b \cap a)) = T \) is the above \( v(a) = T \) and \( v(b) = T \) that is a model for \( \Gamma \rightarrow \Delta \).

It is impossible to find \( v \) which would falsify it, what proves that

\[ \models a, (b \cap a) \rightarrow \neg b, (b \Rightarrow a). \]
Definition of GL

**Axioms of GL:** Any sequent of variables (positive literals) which contains a propositional variable that appears on both sides of the sequent arrow $\rightarrow$, i.e any sequent of the form

$$\Gamma_1', a, \Gamma_2' \rightarrow \Delta_1', a, \Delta_2',$$

for any $a \in VAR$ and any sequences $\Gamma_1', \Gamma_2', \Delta_1', \Delta_2' \in VAR^*$. 

**Inference rules of GL**

We denote by $\Gamma'$, $\Delta'$ finite sequences formed out of literals i.e. out of propositional variables or negations of propositional variables. $\Gamma$, $\Delta$ denote any finite sequences of formulas.
Conjunction rules

\[
(\cap \rightarrow) \quad \frac{\Gamma', A, B, \Gamma \rightarrow \Delta'}{\Gamma', (A \cap B), \Gamma \rightarrow \Delta'}
\]

\[
(\rightarrow \cap) \quad \frac{\Gamma \rightarrow \Delta, A, \Delta'; \Gamma \rightarrow \Delta, B, \Delta'}{\Gamma \rightarrow \Delta, (A \cap B), \Delta'}
\]

Disjunction rules

\[
(\rightarrow \cup) \quad \frac{\Gamma \rightarrow \Delta, A, B, \Delta'}{\Gamma \rightarrow \Delta, (A \cup B), \Delta'}
\]

\[
(\cup \rightarrow) \quad \frac{\Gamma', A, \Gamma \rightarrow \Delta'; \Gamma', B, \Gamma \rightarrow \Delta'}{\Gamma', (A \cup B), \Gamma \rightarrow \Delta'}
\]
Implication rules

$$(\rightarrow \Rightarrow) \quad \frac{\Gamma', A, \Gamma \rightarrow \Delta, B, \Delta'}{\Gamma', \Gamma \rightarrow \Delta, (A \Rightarrow B), \Delta''}.$$ 

$$\quad \Rightarrow \rightarrow \quad \frac{\Gamma', \Gamma \rightarrow \Delta, A, \Delta' \quad ; \quad \Gamma', B, \Gamma \rightarrow \Delta, \Delta'}{\Gamma', (A \Rightarrow B), \Gamma \rightarrow \Delta, \Delta'},$$

Negation rules

$$\quad \neg \rightarrow \quad \frac{\Gamma', \Gamma \rightarrow \Delta, A, \Delta'}{\Gamma', \neg A, \Gamma \rightarrow \Delta, \Delta'},$$

$$\quad \rightarrow \neg \quad \frac{\Gamma', A, \Gamma \rightarrow \Delta, \Delta'}{\Gamma', \Gamma \rightarrow \Delta, \neg A, \Delta'}.$$
We define:

$$GL = (SEQ, AL, (\cup), (\neg\cup), (\cap), (\neg\cap), (\Rightarrow), (\neg\Rightarrow), (\neg\neg))$$

**Formal proof** of a sequent $\Gamma \rightarrow \Delta$ in the proof system $GL$ we understand any sequence

$$\Gamma_1 \rightarrow \Delta_1, \Gamma_2 \rightarrow \Delta_2, \ldots, \Gamma_n \rightarrow \Delta_n$$

of sequents of formulas (elements of $SEQ$), such that $\Gamma_1 \rightarrow \Delta_1 \in AL$, $\Gamma_n \rightarrow \Delta_n = \Gamma \rightarrow \Delta$, and for all $i$ ($1 < i \leq n$) $\Gamma_i \rightarrow \Delta_i \in AL$, or $\Gamma_i \rightarrow \Delta_i$ is a conclusion of one of the inference rules of $GL$ with all its premisses placed in the sequence $\Gamma_1 \rightarrow \Delta_1, \ldots, \Gamma_{i-1} \rightarrow \Delta_{i-1}$.
We write, as usual,

\[\vdash_{\text{GL}} \Gamma \rightarrow \Delta\]
to denote that \(\Gamma \rightarrow \Delta\) has a formal proof in \(\text{GL}\).

A formula \(A \in \mathcal{F}\), has a proof in if the sequent \(\rightarrow A\) has a proof in \(\text{GL}\), i.e. we define:

\[\vdash_{\text{GL}} A \iff \rightarrow A.\]
A proof tree, or GL-proof of $\Gamma \rightarrow \Delta$ is a tree

$$T_{\Gamma \rightarrow \Delta}$$

of sequents satisfying the following conditions:

1. The topmost sequent, i.e. the root of $T_{\Gamma \rightarrow \Delta}$ is $\Gamma \rightarrow \Delta$,

2. All leafs are axioms,

3. The nodes are sequents such that each sequent on the tree follows from the ones immediately preceding it by one of the rules.
We write the proof-trees indicating additionally the name of the inference rule used at each step of the proof.

Remark Proof search, i.e. decomposition tree for a given formula $A$ and hence a proof of $A$ in GL is not always unique!!
For example, a tree-proof (in GL) of the de Morgan law

\( (\neg (a \cap b) \Rightarrow (\neg a \cup \neg b)) \)

is the following.

\[
\begin{align*}
\vdash (\neg (a \cap b) \Rightarrow (\neg a \cup \neg b)) \\
\mid (\Rightarrow \Rightarrow) \\
\neg (a \cap b) \quad \vdash (\neg a \cup \neg b) \\
\mid (\Rightarrow \cup) \\
\neg (a \cap b) \quad \vdash \neg a, \neg b \\
\mid (\Rightarrow \neg) \\
b, \neg (a \cap b) \quad \vdash \neg a \\
\mid (\Rightarrow \neg) \\
b, a, \neg (a \cap b) \quad \vdash \\
\mid (\neg \Rightarrow) \\
b, a \quad \vdash (a \cap b) \\
\bigwedge (\Rightarrow \cap) \\
b, a \quad \vdash a \\
b, a \quad \vdash b
\end{align*}
\]
Exercise 1: Write all other proofs of \( \neg(a \cap b) \Rightarrow (\neg a \cup \neg b) \) in GL.

Exercise 2: Verify that the axiom and the rules of inference of GL are sound, i.e. that the following theorem holds.

**Soundness Theorem** for GL: For any sequent \( \Gamma \rightarrow \Delta \in SEQ \),

if \( \vdash_{GL} \Gamma \rightarrow \Delta \) then \( \models \Gamma \rightarrow \Delta \).

**Completeness Theorem** For any sequent \( \Gamma \rightarrow \Delta \in SEQ \),

\( \vdash_{GL} \Gamma \rightarrow \Delta \) iff \( \models \Gamma \rightarrow \Delta \).

The proof of the Completeness Theorem is similar to the proof for the RS system and is assigned as an exercise.