First Order Predicate Logic
Formal Reasoning

Wolfgang Schreiner and Wolfgang Windsteiger
Wolfgang.(Schreiner|Windsteiger)@risc.jku.at

Research Institute for Symbolic Computation (RISC)
Johannes Kepler University (JKU), Linz, Austria
http://www.risc.jku.at
What is Formal Reasoning?

Problem: how to show that the statement \( \{F_1, \ldots, F_n\} \models G \) is true?

- Is formula \( G \) true in every model in which the \( F_1, \ldots, F_n \) are true?
- \( F_1, \ldots, F_n \): the “axioms” that characterize the considered models.
- \( G \): a “conjecture” that might be also true in all these models.
- If the conjecture is indeed true, then \( G \) is actually a “theorem”.
- Since there are infinitely models, how to ensure that \( G \) is a theorem?

Solution: derive the “sequent” \( F_1, \ldots, F_n \vdash G \) by a proof calculus.

- “Sequent”: a sequence \( F_1, \ldots, F_n \vdash G \) of formulas with “assumptions” \( F_1, \ldots, F_n \) and “goal” \( G \).
- Proof calculus: a set of “inference rules” that derive sequents.
- The inference rules are “sound”: a sequence \( F_1, \ldots, F_n \vdash G \) can only be derived, if \( \{F_1, \ldots, F_n\} \models G \) is true.
- The inference rules are “syntactic”: they only depend on the syntactic structure of the formulas (not their semantics).

Proof: a “proof” is a derivation of a sequent \( F_1, \ldots, F_n \vdash G \).

- Since inference rule are syntactic, the correctness of a proof for goal \( G \) can be mechanically checked (by a human or a computer).
- A correct proof of goal \( G \) turns it from a conjecture to a theorem.

Formal reasoning is the demonstration of truth by checkable proofs.
Formal Reasoning: How to Construct a Proof?

We call sequents $S_1, \ldots, S_n$, and $S$ also proof situations.

Forward interpretation:
A proof starts from trivial proof situations (obviously true),
progresses step-by-step
until it reaches the final situation, where the goal is proved.

Backward interpretation:
A proof starts from the goal to be proved,
until it reaches trivial proof situations (obviously true).

Individual proof steps are guided by inference rules, which are denoted as

\[
\begin{array}{c}
\text{forward} \downarrow & S_1 & \ldots & S_n & \quad \text{backward} \\
\text{S} & & & &
\end{array}
\]

Forward interpretation:
If $S_1, \ldots, S_n$ can be proved,
then also $S$ can be proved.

Backward interpretation:
In order to prove $S$, we need to prove $S_1, \ldots, S_n$.

$S_1, \ldots, S_n$: the premises of the inference rule; $S$: its conclusion.
**Example**

\( S, S_1, \ldots, S_6 \): sequents. Consider the following inference rules:

\[
\begin{align*}
R_1 : & \quad \frac{S_2}{S_1} \quad \frac{S_3}{S_1} \quad \frac{S_4}{S_4} \quad \frac{S_1}{S} \quad \frac{S_5}{S_5} \\
R_2 : & \quad \frac{S_4}{S_4} \quad \frac{S_5}{S_5} \quad \frac{S_6}{S_6} \quad \frac{S_6}{S_3}
\end{align*}
\]

We want to prove \( S \).
Proof Generation vs. Proof Presentation

Proof generation: start with sequent to be proved, then work backwards.

Read and apply rules from bottom to top.

\[
\begin{align*}
R_2: & \quad \frac{S_4}{S} \\
R_5: & \quad \frac{S_2}{S} \\
R_1: & \quad \frac{S_1}{S} \\
R_3: & \quad \frac{S_6}{S_3} \\
R_4: & \quad \frac{S_5}{S} \\
R_6: & \quad \frac{S_6}{S_3} \\
R_7: & \quad \frac{S_6}{S_3}
\end{align*}
\]

Backward style proof presentation: In order to prove \( S \) we have to prove, by \( R_3, S_1 \). For this, by \( R_1 \), we have to

1. prove \( S_2 \): by \( R_5 \) we have to prove \( S_4 \) and \( S_5 \), which are guaranteed by \( R_2 \) and \( R_4 \), respectively. Now we still have to

2. prove \( S_3 \): by \( R_7 \) it is sufficient to prove \( S_6 \), which we know from \( R_6 \).

QED ("quod erat demonstrandum", "what was to be proved").
Proof Generation vs. Proof Presentation

Proof presentation: often done in forward reasoning style, i.e. start with known facts and work forward until the sequent to be proved is reached.

Read and apply rules from top to bottom.

\[
\begin{align*}
R_2 &: S_4 \quad R_4 &: S_5 \\
R_5 &: S_2 \\
R_1 &: S_1 \\
R_3 &: S \\
R_6 &: S_6 \\
R_7 &: S_3
\end{align*}
\]

Forward style proof presentation: We know $S_4$ and $S_5$ can be proved, hence by $R_5$, $S_2$ can be proved. Furthermore we know that $S_6$ can be proved, hence by $R_7$, also $S_3$ can be proved. Together with $S_2$, by $R_1$, we know that $S_1$ can be proved, and therefore, by $R_3$, also $S$. QED.

Proofs are always generated backwards (even if they are later presented in the forward style).
A formal proof can be seen as a tree, where

1. every node is a sequent,
2. if $S_1, \ldots, S_n$ are the children nodes of a node $S$, then there must be an inference rule of the form $\frac{S_1 \ldots S_n}{S}$.

Special case $n = 0$: A leaf has 0 children, hence

for every leaf $S$ in the tree there must be a rule $\frac{}{S}$.

A formal proof of $S$ is a proof tree with root $S$. 
A Sketch of a Simple Proof Generation Procedure

Input: $S$
Output: $P$ such that $P$ is a formal proof of $S$.

$P := \text{tree containing only the root node } S$
$Q := \{S\} \text{ // the leaves of the tree, i.e., the still “open” proof situations}$

\textbf{while} $Q$ not empty
  \begin{itemize}
    \item choose a rule $\frac{S_1 \ldots S_n}{S}$ such that $s \in Q$
    \item replace $s$ in $Q$ by $S_1, \ldots, S_n$
    \item add $S_1, \ldots, S_n$ as children nodes of $s$ in $P$
  \end{itemize}
\textbf{return} $P$

Depending on 1) the rules and 2) the choice of the rule in the loop, the procedure might not terminate or might not give a complete proof.

Proving is the art of selecting the “right” rule applications to elaborate, from the desired goal as a root, a complete proof tree.
Inference Rules: Patterns and Matching

- **Inference rules**: schematic “patterns” to be “matched” against concrete proof situations.

  \[ \frac{K_1 \ldots \vdash G_1 \quad \ldots \quad K_n \ldots \vdash G_n}{K \ldots \vdash G} \]

  - *name:* \( K_1 \ldots \vdash G_1 \quad \ldots \quad K_n \ldots \vdash G_n \)
  - \( K \ldots \vdash G \)
  
  - \( n \) premises and one conclusion (special case \( n = 0 \): no premises).

- **Premises/conclusions**: “patterns” of sequents with schematic variables to be matched by concrete phrases:
  - A “sequence variable” \( (K \ldots) \): an arbitrary sequence of formulas.
  - A “formula variable” \( (F, G, \text{etc.}) \): an arbitrary formula,
  - A “term variable” \( (t, u, \text{etc.}) \): an arbitrary term.

  - Example: \( K \ldots \vdash G_1 \land G_2 \) denotes a sequent whose goal is an arbitrary conjunction whose subformulas are denoted by variables \( G_1 \) and \( G_2 \).

- **Order of assumptions**: does *not* matter.

  - Example: the sequent \( K \ldots, F_1 \land F_2 \vdash G \) indicates that *somewhere* among the assumptions a conjunction occurs.

- **Multiple occurrences of the same variable**: denote the *same* expression everywhere in the inference rule.

  - The sequent \( K \ldots, F \land G \vdash G \) indicates that among the assumptions a conjunction occurs whose second subformula is identical to the goal.
Example: Patterns and Matching

Let \( a \) be a constant, \( f, g \) unary function symbols, \( p, q, r, s \) unary predicate symbols, \( t \) a schematic term variable.

Consider the following inference rules:

\[
R_1: \quad \frac{\vdash r(t)}{\vdash r(f(t))} \quad R_2: \quad \frac{\vdash p(a)}{\vdash s(g(t))} \quad R_3: \quad \frac{\vdash r(f(t))}{\vdash s(t)} \quad R_4: \quad \frac{\vdash p(a)}{\vdash q(a)}
\]

\[
R_5: \quad \frac{\vdash p(t)}{\vdash q(t)} \quad R_6: \quad \frac{\vdash s(f(a))}{\vdash s(t)} \quad R_7: \quad \frac{\vdash s(f(t))}{\vdash s(g(t))}
\]

We want to prove \( \vdash s(a) \).

\[
R_2: \quad \frac{\vdash p(a)}{\vdash q(a)} \quad R_4: \quad \frac{\vdash q(a)}{\vdash s(a)}
\]

\[
R_5: \quad \frac{\vdash r(a)}{\vdash r(f(a))} \quad R_3: \quad \frac{\vdash r(f(a))}{\vdash s(a)} \quad R_6: \quad \frac{\vdash s(f(a))}{\vdash s(g(a))} \quad R_7: \quad \frac{\vdash s(f(t))}{\vdash s(g(t))}
\]
Proof Rules for Predicate Logic

One could give a (minimal) set of inference rules for first order predicate logic, which can be shown to be sound and complete, i.e.

1. every formula, which has a formal proof, is also semantically true and
2. every semantically true formula has a formal proof.

⇝ e.g. sequent calculus, Gentzen calculus, natural deduction calculus, etc.

However, we give proof rules that help in practical proving of mathematical statements and checking of given proofs (differences lie in details only).

We distinguish:

▶ predicate logic rules: equality rules, quantifier rules.

For every logical connective and every standard quantifier, we give at least one rule, where the connective or quantifier occurs as the outermost symbol in the goal or one of the assumptions.
Closing Rules

For closing a proof, we need inference rules without premises.

▶ If the goal is among the assumptions, the goal can be proved.

\[ \text{GoalAssum: } \frac{}{K\ldots, G \vdash G} \]

*Our goal \( G \) occurs among our assumptions.*

▶ If the assumptions are contradictory, any goal can be proved.

\[ \text{ContrAssum: } \frac{}{K\ldots, A, \neg A \vdash G} \]

*From our assumptions we know \( A \) but also \( \neg A \); we have therefore a contradiction and are done with proving \( G \).*

▶ If the assumptions include “false”, any goal can be proved.

\[ \text{FalseAssum: } \frac{}{K\ldots, \bot \vdash G} \]

*Our assumptions include “false”; we are therefore done with proving \( G \).*

The leaves of a proof tree are constructed by application of these rules.
Structural Rules

▶ Any assumption may be dropped:

\[
\text{Drop: } \frac{K \ldots \vdash G}{K \ldots, A \vdash G}
\]

▶ Any assumption may be added, if it is also proved (the “cut rule”):

\[
\text{Cut: } \frac{K \ldots \vdash A \quad K \ldots, A \vdash G}{K \ldots \vdash G}
\]

*We have to prove G. First we prove A: … . Now we prove G with the additional assumption A.*

▶ Rather than proving G, we may assume \( \neg G \) and derive a contradiction (an “indirect proof”):

\[
\text{Indirect: } \frac{K \ldots, \neg G \vdash \bot}{K \ldots \vdash G}
\]

*We have to prove G. Thus we may assume \( \neg G \) and derive a contradiction.*
Connective Rules: Negation

Prove a negation as goal:

\[ P_{\neg}: \quad K \ldots, G \vdash \bot \]

\[ K \ldots \vdash \neg G \]

*We have to prove \( \neg G \). Thus we may assume \( G \) and derive a contradiction.*

Use a negation as an assumption:

\[ A_{\neg}: \quad K \ldots, \neg G \vdash A \]

\[ K \ldots, \neg A \vdash G \]

*We know \( \neg A \) and have to prove \( G \). Thus we may assume \( \neg G \) and prove \( A \).*
We have to prove that $\sqrt{2}$ is not rational. We do a proof by contradiction, hence, we assume that $\sqrt{2}$ was rational and derive a contradiction.
Connective Rules: Conjunction

▶ Prove a conjunction as a goal:

\[
P^-\land: \quad \frac{K \ldots \vdash F_1 \quad K \ldots \vdash F_2}{K \ldots \vdash F_1 \land F_2}
\]

We have to prove \(F_1 \land F_2\). First we prove \(F_1\): …. Now we prove \(F_2\): ….

▶ Use a conjunction as an assumption:

\[
A^-\land: \quad \frac{K \ldots, F_1, F_2 \vdash G}{K \ldots, F_1 \land F_2 \vdash G}
\]

We know \(F_1 \land F_2\), therefore we know \(F_1\) and we know \(F_2\).
Connective Rules: Disjunction

▶ Prove a disjunction as a goal:

\[
\begin{align*}
P-\lor: \quad & K \ldots, \neg F_1 \vdash F_2 \\
\hline
& K \ldots \vdash F_1 \lor F_2 \\
\end{align*}
\]

\[
\begin{align*}
P-\lor: \quad & K \ldots, \neg F_2 \vdash F_1 \\
\hline
& K \ldots \vdash F_1 \lor F_2 \\
\end{align*}
\]

We have to prove \( F_1 \lor F_2 \). Thus we may assume \( \neg F_1 \) and prove \( \neg F_2 \). (or: Thus we may assume \( \neg F_2 \) and prove \( \neg F_1 \)).

▶ Use a disjunction as an assumption ("proof by cases"):

\[
\begin{align*}
A-\lor: \quad & K \ldots, F_1 \vdash G \\
& K \ldots, F_2 \vdash G \\
\hline
& K \ldots, F_1 \lor F_2 \vdash G \\
\end{align*}
\]

We know \( F_1 \lor F_2 \). We proceed by case distinction. Case \( F_1 \): ....
Case \( F_2 \): ....
Example

\[
\begin{align*}
P_1 \quad & \quad P_2 \\
\text{even}(m) \vdash G \quad & \quad \text{odd}(m) \vdash G \\
\text{A-\lor:} \quad & \quad \text{even}(m) \lor \text{odd}(m) \vdash G
\end{align*}
\]

We already know that \( m \) is even or \( m \) is odd. Thus, we can distinguish the two cases:

1. \( m \) is even: \( \ldots \) (insert proof \( P_1 \) here)
2. \( m \) is odd: \( \ldots \) (insert proof \( P_2 \) here)
Connective Rules: Implication

- Prove implication as a goal.

\[
P \rightarrow:\quad \frac{K \ldots, F_1 \vdash F_2}{K \ldots \vdash F_1 \rightarrow F_2}
\]

We have to prove \( F_1 \rightarrow F_2 \). Thus we may assume \( F_1 \) and prove \( F_2 \).

- Use an implication as an assumption:

\[
A \rightarrow:\quad \frac{K \ldots \vdash F_1}{K \ldots, F_1 \rightarrow F_2, F_1 \vdash G}
\]

We know \( F_1 \rightarrow F_2 \). First we prove \( F_1 \): \ldots Now we know \( F_2 \).

- Often used instead: “modus ponens” and “modus tollens”

\[
MP:\quad \frac{K \ldots, F_1, F_2 \vdash G}{K \ldots, F_1 \rightarrow F_2, F_1 \vdash G}
\]

We know \( F_1 \rightarrow F_2 \) and we know \( F_1 \). Therefore we know \( F_2 \).

\[
MT:\quad \frac{K \ldots, \neg F_2, \neg F_1 \vdash G}{K \ldots, F_1 \rightarrow F_2, \neg F_2 \vdash G}
\]

We know \( F_1 \rightarrow F_2 \) and we know \( \neg F_2 \). Therefore we know \( \neg F_1 \).
Example

Prove \([(A \rightarrow (B \lor C)) \land \neg C) \rightarrow (A \rightarrow B)\],

where \(A, B, \) and \(C\) are abbreviations for complex predicate logic formulas.

Develop proof tree top-down with root on top (convenient in practice).

\[
\begin{align*}
P-\rightarrow: & \quad \vdash ((A \rightarrow (B \lor C)) \land \neg C) \rightarrow (A \rightarrow B) \\
A-\land: & \quad (A \rightarrow (B \lor C)) \land \neg C \vdash A \rightarrow B \\
P-\rightarrow: & \quad A \rightarrow (B \lor C), \neg C \vdash A \rightarrow B \\
A-\rightarrow: & \quad \ldots, A \vdash A \\
& \text{GoalAssum:} \quad \ldots, A \vdash A \\
A-\lor: & \quad \neg C, A, B \lor C \vdash B \\
& \text{GoalAssum:} \quad \ldots, B \vdash B \\
& \text{ContrAssum:} \quad \ldots, \neg C, C \vdash B
\end{align*}
\]
Connective Rules: Equivalence

▶ Prove equivalence as a goal:

\[ P \leftrightarrow: \frac{K \ldots \vdash F_1 \to F_2 \quad K \ldots \vdash F_2 \to F_1}{K \ldots \vdash F_1 \leftrightarrow F_2} \]

We have to prove \( F_1 \leftrightarrow F_2 \). First we prove \( F_1 \to F_2 \): .... Now we prove \( F_2 \to F_1 \): ....

▶ Use equivalence as an assumption ("substitution"):

\[ A \leftrightarrow: \frac{K \ldots \vdash F_1 \leftrightarrow F_2 \vdash G}{K \ldots, F_1 \leftrightarrow F_2 \vdash G} \]

\[ A \leftrightarrow: \frac{K \ldots, F_1 \leftrightarrow F_2 \vdash G[F_2/F_1]}{K \ldots, F_1 \leftrightarrow F_2 \vdash G} \]

\[ A \leftrightarrow: \frac{K \ldots \vdash F_1 \leftrightarrow F_2 \vdash G[F_1/F_2]}{K \ldots, F_1 \leftrightarrow F_2 \vdash G} \]

▶ \( \Gamma[F_2/F_1] \): replace in formula(s) \( \Gamma \) some occurrence of \( F_1 \) by \( F_2 \).

We know \( F_1 \leftrightarrow F_2 \) and we know (for example) \( \neg F_2 \land F_3 \). Therefore we know \( \neg F_1 \land F_3 \).
Equality Rules

▶ Prove an equality as a goal:

\[
P-=: \quad K \ldots \vdash t = t
\]

We have to prove \( t = t \) and are therefore done.

▶ Use an equality as assumption (“substitution”):

\[
A-=: \quad \begin{array}{c}
K \ldots [t_2/t_1], t_1 = t_2 \vdash G \\
\hline
K \ldots, t_1 = t_2 \vdash G
\end{array}
\]

\[
A-=: \quad \begin{array}{c}
K \ldots, t_1 = t_2 \vdash G[t_2/t_1] \\
\hline
K \ldots, t_1 = t_2 \vdash G
\end{array}
\]

\[
A-=: \quad \begin{array}{c}
K \ldots [t_1/t_2], t_1 = t_2 \vdash G \\
\hline
K \ldots, t_1 = t_2 \vdash G
\end{array}
\]

\[
A-=: \quad \begin{array}{c}
K \ldots, t_1 = t_2 \vdash G[t_1/t_2] \\
\hline
K \ldots, t_1 = t_2 \vdash G
\end{array}
\]

▶ \( \Gamma[t_2/t_1] \): replace in formula(s) \( \Gamma \) some occurrence of \( t_1 \) by \( t_2 \).

We know \( t_1 = t_2 \) and we know (for example) \( p(a, f(t_1)) \). Therefore we know \( p(a, f(t_2)) \).
We have to prove that \( n \) is even. Since we know \( n = m^2 \), it suffices to prove that \( m^2 \) is even.

\[ \text{A-} =: \frac{\ldots, \text{even}(m), n = m^2 \vdash \text{even}(m^2)}{\ldots, \text{even}(m), n = m^2 \vdash \text{even}(n)} \]
Quantifier Rules: Universal Quantifier

- Prove universally quantified formula as a goal (“skolemization”).
  \[
P-\forall: \frac{K \ldots \vdash F[\bar{x}/x]}{K \ldots \vdash (\forall x : F)} \]
  if \(\bar{x}\) does not occur in \(K \ldots, F\)

We have to prove \((\forall x : p(x, f(x)))\). We take arbitrary but fixed \(\bar{x}\) and prove \(p(\bar{x}, f(\bar{x}))\).

  - “fixed”: “Skolem constant” \(\bar{x}\) in contrast to variable \(x\).
  - “arbitrary”: \(\bar{x}\) is a new constant about which nothing is known (it does not appear anywhere else in the proof situation).

- Use universally quantified formula as an assumption (“instantiation”):
  \[
  A-\forall: \frac{K \ldots, (\forall x : F), F[t/x] \vdash G}{K \ldots, (\forall x : F) \vdash G}
  \]

We know \((\forall x : p(x, f(x)))\). Thus we know (for \(x := a\)) \(p(a, f(a))\) and (for \(x := g(a)\)) \(p(g(a), f(g(a)))\).

  - \((\forall x : F)\) stays in the assumptions and can be instantiated again.
  - A “knowledge generating engine” that can be applied \textit{arbitrarily often}.
  - The problem is to find suitable \(t\) that lets the proof make progress.
  - If an unsuitable \(t\) is chosen, the additional knowledge does not help.
Example

\[ \vdash \text{even}(n) \rightarrow \text{even}(n^2) \]

In order to prove that the square of any even number \( n \) is again even, we take an arbitrary but fixed natural number \( \bar{n} \) and show \( \text{even}(\bar{n}) \rightarrow \text{even}(\bar{n}^2) \).

\[ \vdash (\forall n : \text{even}(n) \rightarrow \text{even}(n^2)) \]

We know that the square of any even number is again even. Hence, this holds for a particular number \( m \) also, i.e. if \( m \) is even then also \( m^2 \) must be even.
Quantifier Rules: Existential Quantifier

- Prove an existentially quantified formula as a goal ("instantiation"):  
  \[ \frac{K \ldots \vdash F[t/x]}{K \ldots \vdash (\exists x : F)} \]

We have to prove \((\exists x : p(x, f(x)))\). We prove (for \(x := g(a)\))  
\(p(g(a), f(g(a)))\).

- The problem is to find a “witness term” \(t\) that lets the proof succeed.
- If an unsuitable \(t\) is chosen, the proof fails.

- Use existentially quantified formula as an assumption ("skolemization"):  
  \[ \frac{K \ldots, F[\overline{x}/x] \vdash G}{K \ldots, (\exists x : F) \vdash G} \quad \text{if} \quad \overline{x} \text{ does not occur in} \quad K \ldots, F, G \]

We know \(\exists x : p(x, f(x))\). Thus we know \(p(\overline{x}, f(\overline{x}))\) for some \(\overline{x}\).

- \(\overline{x}\) is an “arbitrary but fixed” Skolem constant.
- \((\exists x : F)\) disappears from assumptions and cannot be skolemized again.
- A “knowledge generating engine” that can be applied only once.
Example

We have to prove that there exists an $m$ with $2m = 4a$. We prove (for $m := 2a$) $2 \cdot 2a = 4a$.

\[ \forall \exists: \quad \ldots \vdash 2 \cdot 2a = 4a \]
\[ \vdash \exists m : 2m = 4a \]

\[ \forall \exists: \quad \ldots, \frac{m^2}{n^2} = 2 \vdash \ldots \]
\[ \vdash \exists m, n : \frac{m^2}{n^2} = 2 \]

We know there exist $m$ and $n$ such that $\frac{m^2}{n^2} = 2$. Thus we know $\frac{m^2}{n^2} = 2$ for some $\overline{m}$ and $\overline{n}$. 
Example: A Quantifier Proof

We prove

$$\exists x : \forall y : p(x, y) \rightarrow (\forall y : \exists x : p(x, y))$$  \hspace{1cm} (a)

We assume

$$\exists x : \forall y : p(x, y)$$  \hspace{1cm} (1)

and prove

$$\forall y : \exists x : p(x, y)$$  \hspace{1cm} (b)

We take arbitrary but fixed \( y \) and prove

$$\exists x : p(x, y)$$  \hspace{1cm} (c)

From (1), we know (2) \( \forall y : p(\overline{x}, y) \) for some \( \overline{x} \). From (2), we know (for \( y := \overline{y} \)) (3) \( p(\overline{x}, \overline{y}) \). We prove (c) for \( x := \overline{x} \) which we know from (3). QED.
Example: Another Quantifier Proof

We prove

\[((\exists x : p(x)) \land (\forall x : p(x) \rightarrow \exists y : q(x, y))) \rightarrow \exists x, y : q(x, y)\]

(a)

We assume

\((\exists x : p(x)) \land (\forall x : p(x) \rightarrow \exists y : q(x, y))\)

(1)

and show

\exists x, y : q(x, y)

(b)

From (1), we know (2) \((\exists x : p(x))\) and (3) \((\forall x : p(x) \rightarrow \exists y : q(x, y))\). From (2), we know (4) \(p(\bar{x})\) for some \(\bar{x}\). From (3), we know (for \(x := \bar{x}\)) (5) \((p(\bar{x}) \rightarrow \exists y : q(\bar{x}, y))\).

From (4) and (5), we know (6) \(\exists y : q(\bar{x}, y)\). From (6), we know (7) \(q(\bar{x}, \bar{y})\) for some \(\bar{y}\).

We prove (b) for \(x := \bar{x}\) and \(y := \bar{y}\) which we know from (7). QED.
Example: A Quantifier Proof with Branches

We prove

$$
((p(a) \lor q(b)) \land (\forall x : p(x) \rightarrow r(x)) \land (\forall x : q(x) \rightarrow r(f(x)))) \rightarrow \exists x : r(x)
$$

We assume

$$
(p(a) \lor q(b)) \land (\forall x : p(x) \rightarrow r(x)) \land (\forall x : q(x) \rightarrow r(f(x)))
$$

and prove

$$
\exists x : r(x)
$$

From (1), we know (2) \((p(a) \lor q(b)), (3) (\forall x : p(x) \rightarrow r(x)), (4) (\forall x : q(x) \rightarrow r(f(x)))\). From (2), we have two cases:

- Case (5) \(p(a)\): From (3), we know (for \(x := a\)) \((6) p(a) \rightarrow r(a)\). From (5) and (6), we know \((7) r(a)\) and therefore (b) for \(x := a\).
- Case (8) \(q(b)\): From (4), we know (for \(x := b\)) \((9) q(b) \rightarrow r(f(b))\). From (8) and (9), we know \((10) r(f(b))\) and therefore (b) for \(x := f(b)\). QED.
Proving Strategies

- **Proving**: partially “art” but mostly “craft”.
  - Most of a proof is guided by the structure of proof situations.
  - Only in a few places really “creativity” or “ingenuity” is required.
  - Basic strategy: first “spawn” proof tree by the goal-oriented rules and then “close” the branches of the tree by the assumption-oriented rules.

- **First**: apply only the “goal-oriented” rules.
  - Decompose the complex goal into one or more simpler goals.
  - Stop when goals become atomic or existentially quantified.

- **Then**: apply the “assumption-oriented” rules.
  - Decompose complex assumptions into simpler ones.
  - Skolemize existentially quantified assumptions.
  - Instantiate universally quantified assumptions.

- **Ultimately**: “close the gap” between assumptions and goal.
  - How can you derive an atomic goal as an assumption?
  - How can you instantiate an existentially quantified goal such that the instantiated body of the formula appears as an assumption?
  - Try to “match” assumptions and goal by the inference rules.

By considering proving as a “syntactic” process, already a major part of a proof can be elaborated (possibly even completed).
Mathematical Proofs

Mathematical proofs are typically written in a much more informal style.

▶ Do not mention all steps.
▶ Combine several steps into one.
▶ Use for Skolem constant name of variable itself.
▶ Use hidden assumptions. . . .

**Theorem**: Suppose $a$ divides $b$ if and only if, for some $t \in \mathbb{N}$, $b = t \cdot a$. Then, if $a$ divides $b$ it also divides every multiple of $b$.

**Proof**: Assume $a, b, s \in \mathbb{N}$ arbitrary but fixed such that $a$ divides $b$. We show that $a$ divides $s \cdot b$, i.e. $\exists t \in \mathbb{N} : s \cdot b = t \cdot a$. Since $a$ divides $b$, we know $b = \bar{t} \cdot a$ for some $\bar{t} \in \mathbb{N}$, thus, we have to find $t \in \mathbb{N}$ s.t. $s \cdot \bar{t} \cdot a = t \cdot a$. Let now $t := s \cdot \bar{t} \in \mathbb{N}$, we have to show $s \cdot \bar{t} \cdot a = s \cdot t \cdot a$. QED.

Every sentence in the proof is justified by one or more proof rules. Trivial steps (e.g. split conjunction in assumptions) are not mentioned explicitly.

A mathematical proof is an easily readable “sketch” that just gives the essential information to reconstruct a corresponding formal proof.