UNSW SYDNEY

COMP4161
Advanced Topics in Software Verification

$\lambda\rightarrow$ and HOL

Gerwin Klein, June Andronick, Miki Tanaka, Johannes Åman Pohjola

T3/2022

Last time...

- Simply typed lambda calculus $\lambda^{\rightarrow}$
- Typing rules for $\lambda^{\rightarrow}$ [illegible]bles, type contexts
- β-reduction in $\lambda^{\rightarrow}$ [illegible]ject reduction
- β-reduction in $\lambda^{\rightarrow}$ always terminates
- Types and terms in Isabelle

Content

➜ Foundations & Principles

- Intro, Lambda [illegible] atural deduction [1,2]
- Higher Order [illegible] part 1) [2,3[a]]
- Term rewriting [3,4]

➜ Proof & Specification Techniques

- Inductively defined sets, rule induction [4,5]
- Datatype induction, primitive recursion [5,7]
- General recursive functions, termination proofs [7[b]]
- Proof automation, Isar (part 2) [8]
- Hoare logic, proofs about programs, invariants [8,9]
- C verification [9,10]
- Practice, questions, exam prep [10[c]]

[a] a1 due; [b] a2 due; [c] a3 due

Preview: Proofs in Isabelle

Proofs in Isabelle

General schema:

lemma name: "<goal>
apply <method>
apply <method>
. . .
done

Proofs in Isabelle

General schema:

lemma name: "<goal>"
apply <method>
apply <method>
...
done

➜ Sequential application of methods until all **subgoals** are solved.

The Proof State

1. $\bigwedge x_1 \ldots x_p.[\![A_1; \Rightarrow B$
2. $\bigwedge y_1 \ldots y_q.[\![C_1; \Rightarrow D$

The Proof State

1. $\bigwedge x_1 \ldots x_p . [\![A_1; \ldots \Rightarrow B$

2. $\bigwedge y_1 \ldots y_q . [\![C_1; \ldots \Rightarrow D$

$x_1 \ldots x_p$	Parameters
$A_1 \ldots A_n$	Local assumptions
B	Actual (sub)goal

Isabelle Theories

Syntax:

`theory` *MyTh*
`imports` $ImpTh_1$ ...
`begin`
(declarations, definitions, theorems, proofs, ...)*
`end`

- ➜ *MyTh*: name of theory. Must live in file *MyTh*`.thy`
- ➜ $ImpTh_i$: name of *imported* theories. Import transitive.

Isabelle Theories

Syntax:

```
theory MyTh
imports ImpTh1 ...
begin
(declarations, definitions, theorems, proofs, ...)*
end
```

- ➜ *MyTh*: name of theory. Must live in file *MyTh*`.thy`
- ➜ $ImpTh_i$: name of *imported* theories. Import transitive.

Unless you need something special:

```
theory MyTh imports Main begin ... end
```

Natural Deduction Rules

$$\frac{}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B}{C}\ \text{conjE}$$

$$\frac{}{A \vee B}\ \text{disjI}_1 \qquad \frac{}{A \vee B}\ \text{disjI}_2 \qquad \frac{A \vee B}{C}\ \text{disjE}$$

$$\frac{}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):

introduction and **elimination** rules

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B}{C}\ \text{conjE}$$

$$\frac{}{A \vee B}\ \text{disjI}_1 \quad \frac{}{A \vee B}\ \text{disjI}_2 \qquad \frac{A \vee B}{C}\ \text{disjE}$$

$$\frac{}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):
introduction and **elimination** rules

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B \quad [\![A; B]\!] \Longrightarrow C}{C}\ \text{conjE}$$

$$\frac{}{A \vee B}\ \text{disjI}_1 \quad \frac{}{A \vee B}\ \text{disjI}_2 \qquad \frac{A \vee B}{C}\ \text{disjE}$$

$$\frac{}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):
introduction and **elimination** rules

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B \quad [\![A; B]\!] \Longrightarrow C}{C}\ \text{conjE}$$

$$\frac{A}{A \vee B}\ \text{disjI}_1 \quad \frac{B}{A \vee B}\ \text{disjI}_2 \qquad \frac{A \vee B}{C}\ \text{disjE}$$

$$\frac{}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):
introduction and **elimination** rules

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B \quad [\![A; B]\!] \Longrightarrow C}{C}\ \text{conjE}$$

$$\frac{A}{A \vee B}\ \text{disjI}_1 \quad \frac{B}{A \vee B}\ \text{disjI}_2 \qquad \frac{A \vee B \quad A \Longrightarrow C \quad B \Longrightarrow C}{C}\ \text{disjE}$$

$$\frac{}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):
introduction and **elimination** rules

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B \quad [\![A; B]\!] \Longrightarrow C}{C}\ \text{conjE}$$

$$\frac{A}{A \vee B}\ \frac{B}{A \vee B}\ \text{disjI1/2} \qquad \frac{A \vee B \quad A \Longrightarrow C \quad B \Longrightarrow C}{C}\ \text{disjE}$$

$$\frac{A \Longrightarrow B}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):
introduction and **elimination** rules

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B \quad [\![A; B]\!] \Longrightarrow C}{C}\ \text{conjE}$$

$$\frac{A}{A \vee B}\ \frac{B}{A \vee B}\ \text{disjI1, 2} \qquad \frac{A \vee B \quad A \Longrightarrow C \quad B \Longrightarrow C}{C}\ \text{disjE}$$

$$\frac{A \Longrightarrow B}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B \quad A \quad B \Longrightarrow C}{C}\ \text{impE}$$

For each connective ($\wedge$, $\vee$, etc):

introduction and **elimination** rules

Proof by assumption

assumption

proves

1. $[\![B_1; \ldots; B_m]\!] \Longrightarrow C$

by unifying C with one of the B_i

Proof by assumption

assumption

proves

1. $[\![B_1; \ldots; B_m]\!] \Longrightarrow C$

by unifying C with one of the B_i

There may be more than one matching B_i and multiple unifiers.

Backtracking!

Explicit backtracking command: **back**

Intro rules

Intro rules decompose formulas to the right of $\Longrightarrow$.

(e.g. <intro-rule>)

Intro rules

Intro rules decompose formulas to the right of $\Longrightarrow$.

(alternative <intro-rule>)

Intro rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ To prove A it suffices to show $A_1 \ldots A_n$

Intro rules

Intro rules decompose formulas to the right of $\Longrightarrow$.

(also called <intro-rule>)

Intro rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ To prove A it suffices to show $A_1 \ldots A_n$

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Intro rules

Intro rules decompose formulae to the right of $\Longrightarrow$.

(e.g. rule <intro-rule>)

Intro rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ To prove A it suffices to show $A_1 \ldots A_n$

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

➜ unify A and C

➜ replace C with n new subgoals $A_1 \ldots A_n$

Intro rules: example

To prove subgoal $A \longrightarrow A$ can use: $\dfrac{P \Longrightarrow Q}{P \longrightarrow Q}$ impl

(in Isabelle: *impl* : $(?P \Longrightarrow ?Q) \Longrightarrow ?P \longrightarrow ?Q$)

Intro rules: example

To prove subgoal $A \longrightarrow \_$ can use: $\dfrac{P \Longrightarrow Q}{P \longrightarrow Q}$ impl

(in Isabelle: *impl* : $(?P \Longrightarrow ?Q) \Longrightarrow ?P \longrightarrow ?Q$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

- ➜ unify A and C
- ➜ replace C with n new subgoals $A_1 \ldots A_n$

Intro rules: example

To prove subgoal $A \longrightarrow \ldots$ can use: $\dfrac{P \Longrightarrow Q}{P \longrightarrow Q}$ impl

(in Isabelle: *impl* : $(?P \Longrightarrow ?Q) \Longrightarrow ?P \longrightarrow ?Q$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

- ➜ unify A and C
- ➜ replace C with n new subgoals $A_1 \ldots A_n$

Here:

- ➜ unify...
- ➜ replace subgoal...

Intro rules: example

To prove subgoal $A \longrightarrow A$ we can use: $\dfrac{P \Longrightarrow Q}{P \longrightarrow Q}$ impl

(in Isabelle: *impl* : $(?P \Longrightarrow ?Q) \Longrightarrow ?P \longrightarrow ?Q$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

- ➜ unify A and C
- ➜ replace C with n new subgoals $A_1 \ldots A_n$

Here:

- ➜ unify... $?P \longrightarrow ?Q$ with $A \longrightarrow A$
- ➜ replace subgoal...

Intro rules: example

To prove subgoal $A \longrightarrow A$ can use: $\dfrac{P \Longrightarrow Q}{P \longrightarrow Q}$ impl

(in Isabelle: *impl* : $(?P \Longrightarrow ?Q) \Longrightarrow ?P \longrightarrow ?Q$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

- ➜ unify A and C
- ➜ replace C with n new subgoals $A_1 \ldots A_n$

Here:

- ➜ unify... $?P \longrightarrow ?Q$ with $A \longrightarrow A$
- ➜ replace subgoal... $A \longrightarrow A$ (i.e. $[\![\,]\!] \Longrightarrow A \longrightarrow A$)
  with $[\![\, A \,]\!] \Longrightarrow A$ (which can be proved with: **apply** assumption)

Elim rules

Elim rules decompose f[illegible]n the left of $\Longrightarrow$.

a[illegible]le <elim-rule>)

Elim rules

Elim rules decompose f[illegible]n the left of $\Longrightarrow$.

a[illegible]le <elim-rule>)

Elim rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ If I know A_1 and want to prove A it suffices to show $A_2 \ldots A_n$

Elim rules

Elim rules decompose f[illegible]n the left of $\Longrightarrow$.

a[illegible]le <elim-rule>)

Elim rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ If I know A_1 and want to prove A it suffices to show $A_2 \ldots A_n$

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

Elim rules

Elim rules decompose f[illegible]n the left of $\Longrightarrow$.

a[illegible]le <elim-rule>)

Elim rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ If I know A_1 and want to prove A it suffices to show $A_2 \ldots A_n$

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

➜ unifies first premise of rule with an assumption

➜ eliminates that assumption

Elim rules: example

To prove $[\![B \wedge A]\!] \Longrightarrow$ [illegible] use: $\dfrac{P \wedge Q \quad [\![P; Q]\!] \Longrightarrow R}{R}$ conjE

(in Isabelle: *conjE* : $[\![?P \wedge ?Q;\ [\![?P;\ ?Q]\!] \Longrightarrow ?R]\!] \Longrightarrow ?R$)

Elim rules: example

To prove $[\![B \wedge A]\!] \Longrightarrow$? use: $\dfrac{P \wedge Q \quad [\![P; Q]\!] \Longrightarrow R}{R}$ conjE

(in Isabelle: *conjE* : $[\![?P \wedge ?Q;\ [\![?P;\ ?Q]\!] \Longrightarrow ?R]\!] \Longrightarrow ?R$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

- ➜ unifies first premise of rule with an assumption
- ➜ eliminates that assumption

Elim rules: example

To prove $[\![B \wedge A]\!] \Longrightarrow$? use: $\dfrac{P \wedge Q \quad [\![P; Q]\!] \Longrightarrow R}{R}$ conjE

(in Isabelle: *conjE* : $[\![?P \wedge ?Q;\ [\![?P;\ ?Q]\!] \Longrightarrow ?R]\!] \Longrightarrow ?R$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

- ➜ unifies first premise of rule with an assumption
- ➜ eliminates that assumption

Here:

- ➜ unify...
- ➜ and also unify...
- ➜ replace subgoal...

Elim rules: example

To prove $[\![B \wedge A]\!] \Longrightarrow$? use: $\dfrac{P \wedge Q \quad [\![P; Q]\!] \Longrightarrow R}{R}$ conjE

(in Isabelle: *conjE* : $[\![?P \wedge ?Q;\ [\![?P;\ ?Q]\!] \Longrightarrow ?R]\!] \Longrightarrow ?R$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

- ➜ unifies first premise of rule with an assumption
- ➜ eliminates that assumption

Here:

- ➜ unify... $?R$ with A
- ➜ and also unify...
- ➜ replace subgoal...

Elim rules: example

To prove $[\![B \wedge A]\!] \Longrightarrow$? use: $\dfrac{P \wedge Q \quad [\![P; Q]\!] \Longrightarrow R}{R}$ conjE

(in Isabelle: *conjE* : $[\![?P \wedge ?Q;\ [\![?P;\ ?Q]\!] \Longrightarrow ?R]\!] \Longrightarrow ?R$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

- ➜ unifies first premise of rule with an assumption
- ➜ eliminates that assumption

Here:

- ➜ unify... $?R$ with A
- ➜ and also unify... $?P \wedge ?Q$ with assumption $B \wedge A$
- ➜ replace subgoal...

Elim rules: example

To prove $[\![B \wedge A]\!] \Longrightarrow$ ... use: $\dfrac{P \wedge Q \quad [\![P; Q]\!] \Longrightarrow R}{R}$ conjE

(in Isabelle: *conjE* : $[\![?P \wedge ?Q;\ [\![?P;\ ?Q]\!] \Longrightarrow ?R]\!] \Longrightarrow ?R$)

Recall:

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

Like **rule** but also

- ➜ unifies first premise of rule with an assumption
- ➜ eliminates that assumption

Here:

- ➜ unify... $?R$ with A
- ➜ and also unify... $?P \wedge ?Q$ with assumption $B \wedge A$
- ➜ replace subgoal... $[\![B \wedge A]\!] \Longrightarrow A$
  with $[\![B; A]\!] \Longrightarrow A$ (which can be proved with: **apply** assumption)

Demo

More Proof Rules

Iff, Negation, True and False

$$\frac{A = B \qquad A}{B}\ \text{iffE} \qquad \frac{A = B \qquad B}{A}\ \text{iffE}$$

$$\frac{A = B}{A}\ \text{iffD1} \qquad \frac{A = B}{B}\ \text{iffD2}$$

$$\frac{\begin{array}{c}[A]\\ \vdots\\ \bot\end{array}}{\neg A}\ \text{notI} \qquad \frac{\neg A \qquad A}{P}\ \text{notE}$$

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad B}{C}\ \text{iffE}$$

$$\frac{A = B}{A \Longrightarrow B}\ \text{iffD1} \qquad \frac{A = B}{B \Longrightarrow A}\ \text{iffD2}$$

$$\frac{}{\neg A}\ \text{notI} \qquad \frac{\neg A \quad P}{}\ \text{notE}$$

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C}{C}\ \text{iffE}$$

$$\frac{A = B}{}\ \text{iffD1} \qquad \frac{A = B}{}\ \text{iffD2}$$

$$\frac{}{\neg A}\ \text{notI} \qquad \frac{\neg A}{P}\ \text{notE}$$

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C}{C}\ \text{iffE}$$

$$\frac{A = B}{A \Longrightarrow B}\ \text{iffD1} \qquad \frac{A = B}{B \Longrightarrow A}\ \text{iffD2}$$

$$\frac{}{\neg A}\ \text{notI} \qquad \frac{\neg A \quad P}{}\ \text{notE}$$

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C}{C}\ \text{iffE}$$

$$\frac{A = B}{A \Longrightarrow B}\ \text{iffD1} \qquad \frac{A = B}{B \Longrightarrow A}\ \text{iffD2}$$

$$\frac{A \Longrightarrow \mathit{False}}{\neg A}\ \text{notI} \qquad \frac{\neg A}{P}\ \text{notE}$$

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C}{C}\ \text{iffE}$$

$$\frac{A = B}{A \Longrightarrow B}\ \text{iffD1} \qquad \frac{A = B}{B \Longrightarrow A}\ \text{iffD2}$$

$$\frac{A \Longrightarrow \mathit{False}}{\neg A}\ \text{notI} \qquad \frac{\neg A \quad A}{P}\ \text{notE}$$

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C}{C}\ \text{iffE}$$

$$\frac{A = B}{A \Longrightarrow B}\ \text{iffD1} \qquad \frac{A = B}{B \Longrightarrow A}\ \text{iffD2}$$

$$\frac{A \Longrightarrow \mathit{False}}{\neg A}\ \text{notI} \qquad \frac{\neg A \quad A}{P}\ \text{notE}$$

$$\frac{}{\mathit{True}}\ \text{TrueI} \qquad \frac{\mathit{False}}{P}\ \text{FalseE}$$

Equality

$$\frac{}{t = t}\ \text{refl} \qquad \frac{r = s}{s = t}\ \text{sym} \qquad \frac{r = s \quad s = t}{r = t}\ \text{trans}$$

Equality

$$\frac{}{t = t}\ \text{refl} \qquad \frac{s = t}{t = s}\ \text{sym} \qquad \frac{r = s \quad s = t}{r = t}\ \text{trans}$$

$$\frac{s = t \quad P\ s}{P\ t}\ \text{subst}$$

Equality

$$\frac{}{t = t}\ \text{refl} \qquad \frac{s = t}{t = s}\ \text{sym} \qquad \frac{r = s \quad s = t}{r = t}\ \text{trans}$$

$$\frac{s = t \quad P\ s}{P\ t}\ \text{subst}$$

Rarely needed explicitly — used implicitly by term rewriting

Classical

$$\frac{}{P = T \vee P = False}\ \text{True-or-False}$$

Classical

$$\frac{}{P = T \vee P = \mathit{False}}\ \text{True-or-False}$$

excluded-middle

$$\frac{\neg A \Longrightarrow \mathit{False}}{A}\ \text{contr} \qquad \frac{\neg A \Longrightarrow A}{A}\ \text{classical}$$

Classical

$$\frac{}{P = T \vee P = \mathit{False}}\ \text{True-or-False}$$

excluded-middle

$$\frac{\neg A \Longrightarrow \mathit{False}}{A}\ \text{ccontr} \qquad \frac{\neg A \Longrightarrow A}{A}\ \text{classical}$$

➜ **excluded-middle**, **ccontr** and **classical** not derivable from the other rules.

Classical

$$\frac{}{P = T \vee P = \mathit{False}}\ \text{True-or-False}$$

excluded-middle

$$\frac{\neg A \Longrightarrow \mathit{False}}{A}\ \text{ccontr} \qquad \frac{\neg A \Longrightarrow A}{A}\ \text{classical}$$

- **excluded-middle**, **ccontr** and **classical** not derivable from the other rules
- if we include True-or-False, they are derivable

They make the logic "classical", "non-constructive"

Cases

excluded-middle

is a function on type *bool*

Cases

excluded-middle is a function on type *bool*

Isabelle can do case distinctions on arbitrary terms:

apply (case_tac *term*)

Safe and not so safe

Safe rules preserve

Safe and not so safe

Safe rules preserve

conjI, in fl, refl, ccontr, classical, conjE, disjE

$$\frac{A}{A \wedge B}\ \text{conjI}$$

Safe and not so safe

Safe rules preserve provability

conjI, impI, TrueI, refl, ccontr, classical, conjE, disjE

$$\frac{A \quad B}{A \wedge B}\ \text{conjI}$$

Unsafe rules can turn a provable goal into an unprovable one

Safe and not so safe

Safe rules preserve provability

conjI, impI, iffI, refl, ccontr, classical, conjE, disjE

$$\frac{A \quad B}{A \wedge B}\ \text{conjI}$$

Unsafe rules can turn a provable goal into an unprovable one

disjI1, disjI2, impE, iffD1, iffD2, notE

$$\frac{A}{A \vee B}\ \text{disjI1}$$

Safe and not so safe

Safe rules preserve provability

conjI, impI, iffI, refl, ccontr, classical, conjE, disjE

$$\frac{A \quad B}{A \wedge B}\ \text{conjI}$$

Unsafe rules can turn a provable goal into an unprovable one

disjI1, disjI2, impE, iffD1, iffD2, notE

$$\frac{A}{A \vee B}\ \text{disjI1}$$

Apply safe rules before unsafe ones

Demo

What we have learned so far...

- natural deduction r[illegible], $\longrightarrow$, $\neg$, iff...
- proof by assumptio[illegible]ule, elim rule
- safe and unsafe rule[illegible]
- indent your proofs! (one space per subgoal)
- prefer implicit backtracking (chaining) or *rule_tac*, instead of *back*
- *prefer* and *defer*
- *oops* and *sorry*