

# Notations

Notation :

$\Rightarrow$  est noté  $\rightarrow$  ( $\rightarrow$ )

Parenthésage :

$A \rightarrow (B \rightarrow C)$  est simplement noté  $A \rightarrow B \rightarrow C$ .

But courant :

un séquent  $A_1, \dots, A_n \vdash A$  s'affiche

H1 : A1

...

Hn : An

=====

A

# L'environnement de travail

Deux fenêtres : le fichier courant + les réponses de Coq.

L'évaluation du fichier se fait linéairement.

Utilisez les raccourcis **Ctrl+flèches** pour faire avancer/reculer la zone d'évaluation.

```

Emacs: prop.v (/Users/pichardie/Shared/Enseign...
Section Propositional_Logic.

Variable A B C D E F : Prop.

(** Solve them using only [intros] and [apply] *)

Lemma ex01 : (A -> B -> C) -> (A -> B) -> A -> C.
| intros.
Qed.

--:** prop.v Top (8,0) (coq Holes Scripting)-----
| subgoal

A : Prop
B : Prop
C : Prop
D : Prop
E : Prop
F : Prop

=====
(A -> B -> C) -> (A -> B) -> A -> C

--:-- *goals* All (1,0) (CoqGoals Holes)-----

```

# Règle intro $\Rightarrow$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \text{ intro}_{\Rightarrow}$$

=====

A  $\rightarrow$  B

H : A

=====

B

Commande : **intros** H.

## Remarques

- ▶ on peut enchaîner plusieurs introductions : **intros** H1 H2.
- ▶ on peut laisser le système nommer les hypothèses : **intros**.

# Règle $\text{élim}_{\Rightarrow}$

$$\frac{\Gamma \vdash A \Rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B} \text{élim}_{\Rightarrow}$$

$H : A \rightarrow B$   
 =====  
 $B$

$H : A \rightarrow B$   
 =====  
 $A$

Commande : **apply** H.

## Remarques

- ▶ contrairement à  $\text{élim}_{\Rightarrow}$ , il y a un seul sous-arbre de preuve : il faut avoir déjà une preuve de  $A \Rightarrow B$  dans son contexte
- ▶ pour forcer Coq à générer un sous-but pour  $A \Rightarrow B$ , on peut taper **assert**  $(A \rightarrow B)$ .

# Règle Ax

$$\frac{}{\Gamma, A \vdash A} \text{Ax}$$

H : A  
 =====  
 A

Commande : **apply** H.

Remarque

- ▶ on peut éviter de nommer l'hypothèse en utilisant la commande **assumption**.

# Règle intro $\wedge$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \text{ intro}_{\wedge}$$

H : A

H0 : B

=====

A

H : A

H0 : B

=====

A  $\wedge$  B

H : A

H0 : B

=====

B

Commande : **split**.

# Règle $\text{elim}_{\wedge}$

$$\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \text{elim}_{\wedge}^1$$

$$\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \text{elim}_{\wedge}^2$$

H : A  $\wedge$  B

=====

C

H : A

H0 : B

=====

C

Commande : **destruct** H.

## Remarque

- ▶ on peut nommer les hypothèses introduites avec la syntaxe **destruct** H as [H H0].

# Règle $\text{intro}_{\vee}$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \text{intro}_{\vee}^1$$

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \text{intro}_{\vee}^2$$

=====

$A \vee B$

=====

$A$

Commande : **left**.

Remarque

- ▶  $\text{intro}_{\vee}^2$  correspond à la commande **right**.



Règle  $\text{élim}_{\vee}$ 

$$\frac{\Gamma \vdash A \vee B \quad \Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C} \text{élim}_{\vee}$$

H : A

=====

C

H : A  $\vee$  B

=====

C

H : B

=====

C

Commande : **destruct** H.

# Règle $\text{élim}_{\perp}$

$$\frac{\Gamma \vdash \perp}{\Gamma \vdash A} \text{élim}_{\perp}^1$$

H : False

=====

A

Commande : `elim` H.

Remarque

- `elim` H peut aussi s'utiliser quand  $H:A_1 \rightarrow \dots \rightarrow A_n \rightarrow \text{False}$  (mais il faut alors décharger  $n$  sous-buts)

# Règle intro $\neg$

$$\frac{\Gamma, A \vdash \perp}{\Gamma \vdash \neg A} \text{ intro}_{\neg}$$

=====

$\neg A$

H : A

=====

False

Commande : **intros** H.

Remarque

- ▶  $\neg A$  est en fait du sucre syntaxique pour  $A \rightarrow \text{False}$ .
- ▶ **intros**. ne marche pas ici.

# Règle élim $\neg$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash \neg A}{\Gamma \vdash \perp} \text{élim}_{\neg}$$

=====

B

=====

$\neg$  A

=====

A

Commande : **absurd** A.

Remarque

- la commande est utile même quand B n'est pas égal à False.

# Logique du premier ordre en Coq

Notations :

- ▶  $\forall$  se note forall,
- ▶  $\exists$  se note exists

Règle intro $\forall$ 

$$\frac{\Gamma \vdash A}{\Gamma \vdash \forall x A} \text{intro}_{\forall}$$

=====

$\forall x:A, P \ x$

$a : A$

=====

$P \ a$

Commande : **intros** a.

# Règle $\text{elim}_{\forall}$

$$\frac{\Gamma \vdash \forall x A}{\Gamma \vdash A[t/x]} \text{elim}_{\forall}$$

$$\begin{array}{c} H : \forall x:A, P x \\ \hline P a \end{array}$$

Commande : **apply** H.

Remarque

- ▶ si H est de la forme  $\forall x y, Q y \rightarrow P x$ , il faut aider un peu Coq en donnant les substitutions à utiliser : **apply** (H a b) (ici  $x=a$  et  $y=b$ ).

Règle  $\text{intro}_{\exists}$ 

$$\frac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x A} \text{intro}_{\exists}$$

 $a : A$ 

=====

 $\exists x:A, P \ x$ 
 $a : A$ 

=====

 $P \ a$ 

Commande : **exists** a.



# Règle élim $\exists$

$$\frac{\Gamma \vdash \exists x A \quad \Gamma, A \vdash B}{\Gamma \vdash B} \text{élim}_{\exists}$$

H :  $\exists x:A, P \ x$   
 =====  
 B

a : A  
 H : P a  
 =====  
 B

Commande : **destruct** H.

Remarque

- ▶ **destruct** H as [a H]. permet de nommer les objets générés.