

Preuves en logique du premier ordre

David Delahaye

Faculté des Sciences
David.Delahaye@lirmm.fr

Master M1 2018-2019

Logique du premier ordre

Définitions préliminaires

- $\mathcal{V} \equiv$ ensemble de variables d'individu x, y , etc. ;
- $\mathcal{S}_{\mathcal{F}} \equiv$ ensemble de symboles de fonctions f, g , etc. ;
- $\mathcal{S}_{\mathcal{P}} \equiv$ ensemble de symboles de prédicats P, Q , etc. ;
- $\mathcal{S}_{\mathcal{F}} \cap \mathcal{S}_{\mathcal{P}} = \emptyset$;
- Arité $m : \mathcal{S}_{\mathcal{F}} \cup \mathcal{S}_{\mathcal{P}} \rightarrow \mathbb{N}$.

Termes du premier ordre

- Plus petit ensemble \mathcal{T} t.q. :
 - ▶ Si $x \in \mathcal{V}$ alors $x \in \mathcal{T}$;
 - ▶ Si $f \in \mathcal{S}_{\mathcal{F}}$ d'arité n et $t_1, \dots, t_n \in \mathcal{T}$, alors $f(t_1, \dots, t_n) \in \mathcal{T}$.

Logique du premier ordre

Définitions préliminaires

- $\mathcal{V} \equiv$ ensemble de variables d'individu x, y , etc. ;
- $\mathcal{S}_{\mathcal{F}} \equiv$ ensemble de symboles de fonctions f, g , etc. ;
- $\mathcal{S}_{\mathcal{P}} \equiv$ ensemble de symboles de prédicats P, Q , etc. ;
- $\mathcal{S}_{\mathcal{F}} \cap \mathcal{S}_{\mathcal{P}} = \emptyset$;
- Arité $m : \mathcal{S}_{\mathcal{F}} \cup \mathcal{S}_{\mathcal{P}} \rightarrow \mathbb{N}$.

Formules du premier ordre

- Plus petit ensemble \mathcal{F} t.q. :
 - ▶ Si $P \in \mathcal{S}_{\mathcal{P}}$ d'arité n et $t_1, \dots, t_n \in \mathcal{T}$, alors $P(t_1, \dots, t_n) \in \mathcal{F}$;
 - ▶ $\perp, \top \in \mathcal{F}$;
 - ▶ Si $\Phi \in \mathcal{F}$ alors $\neg \Phi \in \mathcal{F}$;
 - ▶ Si $\Phi, \Phi' \in \mathcal{F}$ alors $\Phi \wedge \Phi', \Phi \vee \Phi', \Phi \Rightarrow \Phi' \in \mathcal{F}$;
 - ▶ Si $x \in \mathcal{V}$ et $\Phi \in \mathcal{F}$, alors $\forall x. \Phi, \exists x. \Phi \in \mathcal{F}$.

Logique du premier ordre

Associativité des connecteurs

- \wedge , \vee , et \Leftrightarrow associent à gauche :
 - ▶ $A \wedge B \wedge C \equiv (A \wedge B) \wedge C$.
- \Rightarrow associe à droite :
 - ▶ $A \Rightarrow B \Rightarrow C \equiv A \Rightarrow (B \Rightarrow C)$.

Précédence des connecteurs

- On a la précédence suivante : $\neg \succ \wedge \succ \vee \succ \Rightarrow \succ \Leftrightarrow$;
- Exemples :
 - ▶ $A \wedge B \Rightarrow C \equiv (A \wedge B) \Rightarrow C$;
 - ▶ $A \wedge \neg B \vee C \Rightarrow D \equiv ((A \wedge \neg B) \vee C) \Rightarrow D$;
 - ▶ $A \Rightarrow B \Leftrightarrow C \wedge D \equiv (A \Rightarrow B) \Leftrightarrow (C \wedge D)$.

Notation pointée pour les quantificateurs

- La portée d'un quantificateur va jusqu'à la parenthèse fermante de la formule du quantificateur ;
- Si la formule du quantificateur n'est pas parenthésée, la portée du quantificateur va jusqu'à la fin de la formule ;
- Donc, si on veut arrêter la portée d'un quantificateur, il suffit d'utiliser des parenthèses pour limiter explicitement la portée du quantificateur ;
- Exemple :
 - ▶ $\exists x.P(x) \Rightarrow P(a) \wedge P(b) \equiv \exists x.(P(x) \Rightarrow P(a) \wedge P(b))$;
 - ▶ Si on veut que le \exists ne porte que sur $P(x)$, on doit écrire :
 $(\exists x.P(x)) \Rightarrow P(a) \wedge P(b)$.
- Notation : $\forall x, y. \Phi \equiv \forall x. \forall y. \Phi$ (idem pour \exists).

Logique classique

- Une formule est toujours vraie ou fausse ;
- Que je puisse en démontrer la validité ou non ;
- Logique bi-valuée (vrai, faux) ;
- Logique du « tiers exclu » : $A \vee \neg A$.

Logique intuitionniste ou constructive

- Une formule est vraie, fausse, ou « on ne sait pas » ;
- Si on ne sait en démontrer la validité, alors « on ne sait pas » ;
- Logique tri-valuée d'une certaine manière ;
- Le « tiers exclu » n'est pas admis dans cette logique.

Systèmes de preuves

Plusieurs systèmes

- Systèmes à la Frege-Hilbert ;
- Systèmes à la Gentzen :
 - ▶ Dédution naturelle ;
 - ▶ Calcul des séquents.

Adéquation vis-à-vis de la sémantique

- Correction et complétude par rapport à la sémantique ;
- Correction : si je trouve une preuve de P alors P est vraie ;
- Complétude : si P est vraie alors il existe une preuve de P ;
- Preuve \equiv moyen syntaxique de vérifier la validité d'une formule.

Calcul des séquents intuitionniste

Règles

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

$$\frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B} \text{ cont}$$

$$\frac{\Gamma \vdash A \quad \Gamma, B \vdash C}{\Gamma, A \Rightarrow B \vdash C} \Rightarrow_{\text{left}}$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \Rightarrow_{\text{right}}$$

$$\frac{\Gamma \vdash A \quad \Gamma, B \vdash C}{\Gamma, A \Leftrightarrow B \vdash C} \Leftrightarrow_{\text{left1}}$$

$$\frac{\Gamma \vdash B \quad \Gamma, A \vdash C}{\Gamma, A \Leftrightarrow B \vdash C} \Leftrightarrow_{\text{left2}}$$

$$\frac{\Gamma, A \vdash B \quad \Gamma, B \vdash A}{\Gamma \vdash A \Leftrightarrow B} \Leftrightarrow_{\text{right}}$$

Calcul des séquents intuitionniste

Règles

$$\frac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C} \wedge_{\text{left}}$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge_{\text{right}}$$

$$\frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma, A \vee B \vdash C} \vee_{\text{left}}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee_{\text{right1}}$$

$$\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee_{\text{right2}}$$

$$\frac{\Gamma \vdash A}{\Gamma, \neg A \vdash B} \neg_{\text{left}}$$

$$\frac{\Gamma, A \vdash \perp}{\Gamma \vdash \neg A} \neg_{\text{right}}$$

$$\frac{}{\Gamma, \perp \vdash A} \perp_{\text{left}}$$

$$\frac{}{\Gamma \vdash \top} \top_{\text{right}}$$

Calcul des séquents intuitionniste

Règles

$$\frac{\Gamma, A(t) \vdash B}{\Gamma, \forall x A(x) \vdash B} \forall_{\text{left}}$$

$$\frac{\Gamma \vdash A(x)}{\Gamma \vdash \forall x A(x)} \forall_{\text{right}}, x \notin \Gamma$$

$$\frac{\Gamma, A(x) \vdash B}{\Gamma, \exists x.A(x) \vdash B} \exists_{\text{left}}, x \notin \Gamma, B$$

$$\frac{\Gamma \vdash A(t)}{\Gamma \vdash \exists x.A(x)} \exists_{\text{right}}$$

$$\frac{\Gamma \vdash A \quad \Gamma, A \vdash B}{\Gamma \vdash B} \text{cut}$$

Calcul des séquents classique

Règles

$$\frac{\Gamma, A(t) \vdash B}{\Gamma, \forall x A(x) \vdash B} \forall_{\text{left}}$$

$$\frac{\Gamma \vdash A(x)}{\Gamma \vdash \forall x A(x)} \forall_{\text{right}}, x \notin \Gamma$$

$$\frac{\Gamma, A(x) \vdash B}{\Gamma, \exists x.A(x) \vdash B} \exists_{\text{left}}, x \notin \Gamma, B$$

$$\frac{\Gamma \vdash A(t)}{\Gamma \vdash \exists x.A(x)} \exists_{\text{right}}$$

$$\frac{\Gamma \vdash A \quad \Gamma, A \vdash B}{\Gamma \vdash B} \text{cut}$$

$$\frac{\Gamma \vdash \neg\neg A}{\Gamma \vdash A} \text{em}$$

Calcul des séquents classique (LK)

Règles

$$\frac{}{\Gamma, A \vdash \Delta, A} \text{ ax}$$

$$\frac{\Gamma \vdash A \quad \Gamma, A \vdash B}{\Gamma \vdash B} \text{ cut}$$

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta} \text{ cont}_{\text{left}}$$

$$\frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A} \text{ cont}_{\text{right}}$$

Calcul des séquents classique (LK)

Règles

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow_{\text{left}} \qquad \frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B} \Rightarrow_{\text{right}}$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma, B \vdash \Delta}{\Gamma, A \Leftrightarrow B \vdash \Delta} \Leftrightarrow_{\text{left1}}$$

$$\frac{\Gamma \vdash \Delta, B \quad \Gamma, A \vdash \Delta}{\Gamma, A \Leftrightarrow B \vdash \Delta} \Leftrightarrow_{\text{left2}}$$

$$\frac{\Gamma, A \vdash \Delta, B \quad \Gamma, B \vdash \Delta, A}{\Gamma \vdash \Delta, A \Leftrightarrow B} \Leftrightarrow_{\text{right}}$$

Calcul des séquents classique (LK)

Règles

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge_{\text{left}}$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B} \wedge_{\text{right}}$$

$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee_{\text{left}}$$

$$\frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B} \vee_{\text{right}}$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \neg_{\text{left}}$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \neg_{\text{right}}$$

$$\frac{}{\Gamma, \perp \vdash \Delta} \perp_{\text{left}}$$

$$\frac{}{\Gamma \vdash \Delta, \top} \top_{\text{right}}$$

Calcul des séquents classique (LK)

Règles

$$\frac{\Gamma, A(t) \vdash \Delta}{\Gamma, \forall x.A(x) \vdash \Delta} \forall_{\text{left}}$$

$$\frac{\Gamma \vdash \Delta, A(x)}{\Gamma \vdash \Delta, \forall x.A(x)} \forall_{\text{right}}, x \notin \Gamma, \Delta$$

$$\frac{\Gamma, A(x) \vdash \Delta}{\Gamma, \exists x.A(x) \vdash \Delta} \exists_{\text{left}}, x \notin \Gamma, \Delta$$

$$\frac{\Gamma \vdash \Delta, A(t)}{\Gamma \vdash \Delta, \exists x.A(x)} \exists_{\text{right}}$$

Exemple de preuve au premier ordre dans LJ/LK

Négation et quantificateurs

$$\frac{\frac{\frac{\overline{P(x) \vdash P(x)}}{P(x) \vdash \exists x.P(x)} \exists_{\text{right}}}{\neg(\exists x.P(x)), P(x) \vdash \perp} \neg_{\text{left}}}{\neg(\exists x.P(x)) \vdash \neg P(x)} \neg_{\text{right}}}{\neg(\exists x.P(x)) \vdash \forall x.\neg P(x)} \forall_{\text{right}}}{\vdash \neg(\exists x.P(x)) \Rightarrow \forall x.\neg P(x)} \Rightarrow_{\text{right}}$$

Logiques classique/intuitionniste

Sémantique du « il existe »

- En logique classique : $\exists x.P(x) \equiv$ il existe n termes t_1, t_2, \dots, t_n tels que $P(t_1) \vee P(t_2) \vee \dots \vee P(t_n)$ est vraie (théorème de Herbrand) ;
- En logique intuitionniste : $\exists x.P(x) \equiv$ il existe un terme t tel que $P(t)$ est vraie.

On doit construire un témoin t qui vérifie P et en avoir l'intuition.
D'où le nom de logique « intuitionniste » ou « constructive ».

Logique classique

- La logique classique est une logique assez « exotique » ;
- On peut démontrer une formule $\exists x.P(x)$ sans jamais montrer un seul témoin qui fonctionne (c'est-à-dire qui vérifie P) !
- De ce fait, c'est plus facile de faire des preuves en logique classique qu'en logique intuitionniste.

Exemple de preuve en logique classique

Petit théorème mathématique

- Il existe a et b irrationnels tels que a^b est rationnel ;
- Preuve :
 - ▶ Utilisation du tiers exclu : $\sqrt{2}^{\sqrt{2}}$ est rationnel ou non ; deux cas :
 - ★ Si $\sqrt{2}^{\sqrt{2}}$ est rationnel, alors le théorème est vrai ;
 - ★ Si $\sqrt{2}^{\sqrt{2}}$ est irrationnel, alors $(\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = \sqrt{2}^2 = 2$, qui est rationnel.

En logique intuitionniste

- Le théorème est vrai en logique intuitionniste ;
- Mais on doit montrer un a et b qui fonctionnent ;
- Plusieurs pages de théorie des nombres non triviales !

Un autre exemple de preuve en logique classique

Preuve dans LK

Démontrer : $\exists x.P(x) \Rightarrow P(a) \wedge P(b)$.

Cette formule est-elle valide ?

Un autre exemple de preuve en logique classique

Preuve dans LK

$$\frac{\frac{\frac{\frac{\frac{\Gamma \vdash P(a), P(a) \wedge P(b)}{\Gamma \vdash P(a), P(b)} \text{ ax} \quad \frac{\Gamma \vdash P(b), P(a) \wedge P(b)}{\Gamma \vdash P(b)} \text{ ax}}{\Gamma \vdash P(a), P(b), P(a) \wedge P(b)} \wedge_{\text{right}}}{\Gamma \vdash P(a), P(b) \vdash P(a) \wedge P(b), P(a) \wedge P(b)} \Rightarrow_{\text{right}}}{\frac{P(a) \vdash P(a) \wedge P(b), P(b) \Rightarrow P(a) \wedge P(b)}{P(a) \vdash P(a) \wedge P(b), \exists x.P(x) \Rightarrow P(a) \wedge P(b)} \exists_{\text{right}}}{\frac{P(a) \vdash P(a) \wedge P(b), \exists x.P(x) \Rightarrow P(a) \wedge P(b)}{\vdash P(a) \Rightarrow P(a) \wedge P(b), \exists x.P(x) \Rightarrow P(a) \wedge P(b)} \Rightarrow_{\text{right}}}{\frac{\vdash P(a) \Rightarrow P(a) \wedge P(b), \exists x.P(x) \Rightarrow P(a) \wedge P(b)}{\vdash \exists x.P(x) \Rightarrow P(a) \wedge P(b), \exists x.P(x) \Rightarrow P(a) \wedge P(b)} \exists_{\text{right}}}{\vdash \exists x.P(x) \Rightarrow P(a) \wedge P(b)} \text{ cont}_{\text{right}}$$

Exercices en logique propositionnelle

Propositions à démontrer

- ❶ $A \Rightarrow B \Rightarrow A$
- ❷ $(A \Rightarrow B \Rightarrow C) \Rightarrow (A \Rightarrow B) \Rightarrow A \Rightarrow C$
- ❸ $A \wedge B \Rightarrow B$
- ❹ $B \Rightarrow A \vee B$
- ❺ $(A \vee B) \Rightarrow (A \Rightarrow C) \Rightarrow (B \Rightarrow C) \Rightarrow C$
- ❻ $A \Rightarrow \perp \Rightarrow \neg A$
- ❼ $\perp \Rightarrow A$
- ❽ $(A \Leftrightarrow B) \Rightarrow A \Rightarrow B$
- ❾ $(A \Leftrightarrow B) \Rightarrow B \Rightarrow A$
- ❿ $(A \Rightarrow B) \Rightarrow (B \Rightarrow A) \Rightarrow (A \Leftrightarrow B)$

Exercices en logique du premier ordre

Propositions à démontrer

- ❶ $\forall x. P(x) \Rightarrow \exists y. P(y) \vee Q(y)$
- ❷ $(\exists x. P(x) \vee Q(x)) \Rightarrow (\exists x. P(x)) \vee (\exists x. Q(x))$
- ❸ $(\forall x. P(x)) \wedge (\forall x. Q(x)) \Rightarrow \forall x. P(x) \wedge Q(x)$
- ❹ $(\forall x. P(x) \wedge Q(x)) \Rightarrow (\forall x. P(x)) \wedge (\forall x. Q(x))$
- ❺ $(\forall x. \neg P(x)) \Rightarrow \neg(\exists x. P(x))$
- ❻ $\neg(\forall x. P(x)) \Rightarrow \exists x. \neg P(x)$

Outil d'aide à la preuve Coq

Caractéristiques

- Développement par l'équipe Inria πr^2 ;
- Preuve de programmes fonctionnels ;
- Théorie des types (calcul des constructions inductives) ;
- Isomorphisme de Curry-Howard (objets preuves).

Implantation

- Premières versions milieu des années 80 ;
- Implantation actuelle en OCaml ;
- Preuve interactive (peu d'automatisation) ;
- En ligne de commande ou avec l'interface graphique CoqIDE.

Pour les séances de TP

- Installer Coq : <https://coq.inria.fr/>.

Logique propositionnelle

Exemples de preuves

- Implication :

```
Coq < Parameter A : Prop.
```

```
A is assumed
```

```
Coq < Goal A -> A.
```

```
1 subgoal
```

```
=====
```

```
A -> A
```

Logique propositionnelle

Exemples de preuves

- Implication :

```
Coq < intro.
```

```
1 subgoal
```

```
H : A
```

```
=====
```

```
A
```

Exemples de preuves

- Implication :

```
Coq < assumption.
```

```
No more subgoals.
```

```
Coq < Save my_thm.
```

```
intro.
```

```
assumption.
```

```
my_thm is defined
```

Logique propositionnelle

Exemples de preuves

- Application (modus ponens) :

```
Coq < Parameters A B : Prop.
```

```
A is assumed
```

```
B is assumed
```

```
Coq < Goal (A -> B) -> A -> B.
```

```
1 subgoal
```

```
=====
```

```
(A -> B) -> A -> B
```

Exemples de preuves

- Application (modus ponens) :

```
Coq < intros.
```

```
1 subgoal
```

```
H : A -> B
```

```
H0 : A
```

```
=====
```

```
B
```

```
Coq < apply (H H0).
```

```
No more subgoals.
```

Exemples de preuves

- Connecteurs \wedge et \vee :

```
Coq < Parameters A B : Prop.
```

```
A is assumed
```

```
B is assumed
```

```
Coq < Goal A /\ B -> A.
```

```
1 subgoal
```

```
=====
```

```
A /\ B -> A
```

Logique propositionnelle

Exemples de preuves

- Connecteurs \wedge et \vee :

```
Coq < intro.
```

```
1 subgoal
```

```
H : A /\ B
```

```
=====
```

```
A
```


Logique propositionnelle

Exemples de preuves

- Connecteurs \wedge et \vee :

```
Coq < elim H.
```

```
1 subgoal
```

```
H : A /\ B
```

```
=====
```

```
A -> B -> A
```

Logique propositionnelle

Exemples de preuves

- Connecteurs \wedge et \vee :

```
Coq < intros.
```

```
1 subgoal
```

```
H : A /\ B
```

```
H0 : A
```

```
H1 : B
```

```
=====
```

```
A
```

```
Coq < assumption.
```

```
No more subgoals.
```

Logique propositionnelle

Exemples de preuves

- Connecteurs \wedge et \vee :

```
Coq < Parameters A B : Prop.
```

```
A is assumed
```

```
B is assumed
```

```
Coq < Goal A -> A  $\vee$  B.
```

```
1 subgoal
```

```
=====
```

```
A -> A  $\vee$  B
```

Logique propositionnelle

Exemples de preuves

- Connecteurs \wedge et \vee :

Coq < intro.

1 subgoal

H : A

=====

A \vee B

Exemples de preuves

- Connecteurs \wedge et \vee :

```
Coq < left.
```

```
1 subgoal
```

```
H : A
```

```
=====
```

```
A
```

```
Coq < assumption.
```

```
No more subgoals.
```

Logique propositionnelle

Exemples de preuves

- Connecteurs \neg :

```
Coq < Parameters A B : Prop.
```

```
A is assumed
```

```
B is assumed
```

```
Coq < Goal A -> ~A -> False.
```

```
1 subgoal
```

```
=====
```

```
A -> ~ A -> False
```

Exemples de preuves

- Connecteurs \neg :

```
Coq < intros.
```

```
1 subgoal
```

```
  H : A
```

```
  H0 : ~ A
```

```
=====
```

```
  False
```

```
Coq < apply (H0 H).
```

```
No more subgoals.
```

Propositions à démontrer

- ❶ $A \Rightarrow B \Rightarrow A$
- ❷ $(A \Rightarrow B \Rightarrow C) \Rightarrow (A \Rightarrow B) \Rightarrow A \Rightarrow C$
- ❸ $A \wedge B \Rightarrow B$
- ❹ $B \Rightarrow A \vee B$
- ❺ $(A \vee B) \Rightarrow (A \Rightarrow C) \Rightarrow (B \Rightarrow C) \Rightarrow C$
- ❻ $A \Rightarrow \perp \Rightarrow \neg A$
- ❼ $\perp \Rightarrow A$
- ❽ $(A \Leftrightarrow B) \Rightarrow A \Rightarrow B$
- ❾ $(A \Leftrightarrow B) \Rightarrow B \Rightarrow A$
- ❿ $(A \Rightarrow B) \Rightarrow (B \Rightarrow A) \Rightarrow (A \Leftrightarrow B)$

Logique du premier ordre

Exemples de preuves

- Quantificateur \forall :

```
Coq < Parameter E : Set.
```

```
E is assumed
```

```
Coq < Parameter P : E -> Prop.
```

```
P is assumed
```

```
Coq < Goal forall x : E, (P x) -> (P x).
```

```
1 subgoal
```

```
=====
```

```
forall x : E, P x -> P x
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \forall :

Coq < intros.

1 subgoal

x : E

H : P x

=====

P x

Coq < assumption.

No more subgoals.

Logique du premier ordre

Exemples de preuves

- Quantificateur \forall :

```
Coq < Parameter E : Set.
```

```
E is assumed
```

```
Coq < Parameter a : E.
```

```
a is assumed
```

```
Coq < Parameter P : E -> Prop.
```

```
P is assumed
```

```
Coq < Goal (forall x : E, (P x)) -> (P a).
```

```
1 subgoal
```

```
=====
```

```
(forall x : E, P x) -> P a
```

Exemples de preuves

- Quantificateur \forall :

```
Coq < intro.
```

```
1 subgoal
```

```
H : forall x : E, P x
```

```
=====
```

```
P a
```

```
Coq < apply H.
```

```
No more subgoals.
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < Parameter E : Set.
```

```
E is assumed
```

```
Coq < Parameter a : E.
```

```
a is assumed
```

```
Coq < Parameter P : E -> Prop.
```

```
P is assumed
```

```
Coq < Goal (P a) -> exists x : E, (P x).
```

```
1 subgoal
```

```
=====
```

```
P a -> exists x : E, P x
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < intro.
```

```
1 subgoal
```

```
H : P a
```

```
=====
```

```
exists x : E, P x
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

Coq < exists a.

1 subgoal

H : P a

=====

P a

Coq < assumption.

No more subgoals.

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < Parameter E : Set.
```

```
E is assumed
```

```
Coq < Parameter a : E.
```

```
a is assumed
```

```
Coq < Parameter P : E -> Prop.
```

```
P is assumed
```

```
Coq < Goal (exists x : E, ~(P x)) ->  
          ~(forall x : E, (P x)).
```

```
1 subgoal
```

```
=====
```

```
(exists x : E, ~ P x) -> ~ (forall x : E, P x)
```


Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < intros.
```

```
1 subgoal
```

```
H : exists x : E, ~ P x
```

```
=====
```

```
~ (forall x : E, P x)
```

```
Coq < red.
```

```
1 subgoal
```

```
H : exists x : E, ~ P x
```

```
=====
```

```
(forall x : E, P x) -> False
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < intro.
```

```
1 subgoal
```

```
H : exists x : E, ~ P x
```

```
H0 : forall x : E, P x
```

```
=====
```

```
False
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < elim H.
```

```
1 subgoal
```

```
H : exists x : E, ~ P x
```

```
H0 : forall x : E, P x
```

```
=====
```

```
forall x : E, ~ P x -> False
```

Exemples de preuves

- Quantificateur \exists :

```
Coq < intros.
```

```
1 subgoal
```

```
H : exists x : E, ~ P x
```

```
H0 : forall x : E, P x
```

```
x : E
```

```
H1 : ~ P x
```

```
=====
```

```
False
```

Logique du premier ordre

Exemples de preuves

- Quantificateur \exists :

```
Coq < apply H1.
```

```
1 subgoal
```

```
H : exists x : E, ~ P x
```

```
H0 : forall x : E, P x
```

```
x : E
```

```
H1 : ~ P x
```

```
=====
```

```
P x
```

```
Coq < apply H0.
```

```
No more subgoals.
```

Propositions à démontrer

- ❶ $\forall x. P(x) \Rightarrow \exists y. P(y) \vee Q(y)$
- ❷ $(\exists x. P(x) \vee Q(x)) \Rightarrow (\exists x. P(x)) \vee (\exists x. Q(x))$
- ❸ $(\forall x. P(x)) \wedge (\forall x. Q(x)) \Rightarrow \forall x. P(x) \wedge Q(x)$
- ❹ $(\forall x. P(x) \wedge Q(x)) \Rightarrow (\forall x. P(x)) \wedge (\forall x. Q(x))$
- ❺ $(\forall x. \neg P(x)) \Rightarrow \neg(\exists x. P(x))$
- ❻ $\neg(\forall x. P(x)) \Rightarrow \exists x. \neg P(x)$

Guide de survie du petit Coq-uin

Correspondance LK/Coq

Logique propositionnelle		Logique du premier ordre	
Règle LK	Tactique Coq	Règle LK	Tactique Coq
ax	assumption	\forall_{right}	intro
cut	cut	\forall_{left}	apply
$\Rightarrow_{\text{right}}$	intro	\exists_{right}	exists
$\Rightarrow_{\text{left}}$	apply	\exists_{left}	elim
$\Leftrightarrow_{\text{right}}$	split		
$\Leftrightarrow_{\text{left}}$	elim		
\wedge_{right}	split		
\wedge_{left}	elim		
\vee_{right1}	left		
\vee_{right2}	right		
\vee_{left}	elim		
\neg_{right}	intro		
\neg_{left}	elimtype False + apply		
$\top_{\text{right}}, \perp_{\text{left}}$	auto		