

## Preuves assistées par ordinateur – TD n° 10

## Théorie des types simples et système T

**Exercice 1 (Système T)** On rappelle qu'il système T de Gödel est l'extension du  $\lambda$ -calcul simplement typé (à un seul type de base  $\text{Nat}$ ) obtenu en ajoutant aux termes<sup>1</sup> :

- Des constantes  $0 : \text{Nat}$  et  $S : \text{Nat} \rightarrow \text{Nat}$  (les « constructeurs »);
- Pour chaque type simple, un constant  $\text{rec}^\tau : \rightarrow (\text{Nat} \rightarrow \rightarrow) \rightarrow \text{Nat} \rightarrow$  (le « récursur » associé au type) muni des règles de réduction :

$$\begin{array}{ll} \text{rec}^\tau x f 0 & x \\ \text{rec}^\tau x f (S n) & f n (\text{rec}^\tau x f n) \end{array}$$

1. Implémenter dans le système T l'addition, la multiplication, la fonction prédécesseur, le test de nullité, la fonction puissance et la factorielle.
2. Écrire dans le système T un terme  $\text{ack} : \text{Nat} \rightarrow \text{Nat} \rightarrow \text{Nat}$  qui implémente la fonction d'Ackermann définie par les équations :

$$\begin{array}{ll} \text{ack}(0; m) & = m + 1 & (m \geq 0) \\ \text{ack}(n; 0) & = \text{ack}(n - 1; 1) & (n > 0) \\ \text{ack}(n; m) & = \text{ack}(n - 1; \text{ack}(n; m - 1)) & (n; m > 0) \end{array}$$

**Exercice 2 (Codage imprédicatif des connecteurs)** En théorie des types simples, les unités  $\top$ ,  $\perp$  et les connecteurs  $\wedge$ ,  $\vee$  sont définis par

$$\begin{array}{ll} \top \equiv \forall x : o : (x \Rightarrow x) & A \wedge B \equiv \forall x : o : ((A \Rightarrow B \Rightarrow x) \Rightarrow x) \\ \perp \equiv \forall x : o : x & A \vee B \equiv \forall x : o : ((A \Rightarrow x) \Rightarrow (B \Rightarrow x) \Rightarrow x) \end{array}$$

Montrer que les règles suivantes sont admissibles :

$$\begin{array}{c} \frac{\Sigma \vdash \Gamma \text{ ctx}}{\Gamma \vdash_\Sigma \top} \qquad \frac{\Gamma \vdash_\Sigma \perp \quad \Sigma \vdash A : o}{\Gamma \vdash_\Sigma A} \\[10pt] \frac{\Gamma \vdash_\Sigma A \quad \Gamma \vdash_\Sigma B}{\Gamma \vdash_\Sigma A \wedge B} \qquad \frac{\Gamma \vdash_\Sigma A \wedge B}{\Gamma \vdash_\Sigma A} \text{ (+ règle symétrique)} \\[10pt] \frac{\Gamma \vdash_\Sigma A \quad \Sigma \vdash B : o}{\Gamma \vdash_\Sigma A \vee B} \text{ (+ règle symétrique)} \qquad \frac{\Gamma; A \vdash_\Sigma C \quad \Gamma; B \vdash_\Sigma C \quad \Gamma \vdash_\Sigma A \vee B}{\Gamma \vdash_\Sigma C} \end{array}$$

*Indication :* On pourra utiliser la règle admissible d'affaiblissement (au sens de la logique).

---

1. En réalité, le système T comporte également des types produit ( $\times$ ) et somme ( $+$ ) munis des constructions associées au niveau des termes, mais nous n'en aurons pas besoin dans le cadre de cet exercice.

**Exercice 3 (Quantificateur existentiel)** Pour toute proposition  $A$  dépendant (éventuellement) d'un variable  $x$  : on not

$$\exists x : A \equiv \forall p : o : [\forall x : (A \Rightarrow p) \Rightarrow p] :$$

Montrer que les règles suivantes sont dérivables :

$$\frac{\Sigma; [x : ] \vdash A : o \quad \Sigma \vdash N : \quad \Gamma \vdash_{\Sigma} A\{x := N\}}{\Gamma \vdash_{\Sigma} \exists x : A}$$

$$\frac{\Gamma; A \vdash_{\Sigma} B \quad \Gamma \vdash_{\Sigma} \exists x : A}{\Gamma \vdash_{\Sigma} B} \quad (x \notin FV(\Gamma, B))$$

**Exercice 4 (Égalité de Leibniz)** Étant donnés deux termes  $M_1$  et  $M_2$  de type  $\tau$ , on not

$$M_1 =_{\tau} M_2 \equiv \forall p : (\rightarrow o) : (p M_1 \Rightarrow p M_2) :$$

Montrer que les propositions suivantes sont dérivables :

$$\begin{aligned} \forall x : & \quad (x = x) \\ \forall x : \forall y : & \quad (x = y \Rightarrow y = x) \\ \forall x : \forall y : \forall z : & \quad (x = y \Rightarrow y = z \Rightarrow x = z) \\ \forall x : \forall y : & \quad (x = y \Rightarrow \forall p : \rightarrow o : (p x \Rightarrow p y)) \end{aligned}$$

*Indication :* On commencera d'abord par chercher un *pruv informelle*.

**Exercice 5 (Arithmétique de Peano)** Afin de pouvoir raisonner sur les entiers naturels dans la théorie des types simples (dans le type  $\rightarrow$ ), on introduit deux constantes  $0 : \tau$  et  $S : \rightarrow$  (au niveau des termes du  $\lambda$ -calcul simple non typé) et on ajoute aux règles de déduction les deux axiomes suivants

$$\frac{\Sigma \vdash \Gamma \text{ ctx}}{\Gamma \vdash_{\Sigma} \forall x : \forall y : (S x =_{\iota} S y \Rightarrow x =_{\iota} y)} \quad \frac{\Sigma \vdash \Gamma \text{ ctx}}{\Gamma \vdash_{\Sigma} \forall x : (S x =_{\iota} 0 \Rightarrow \perp)}$$

correspondant aux principes habituels d'injectivité et de non-confusion (en utilisant l'égalité de Leibniz «  $=_{\iota}$  » défini à l'exercice précédent).

1. Écrire une proposition qui exprime le schéma de récurrence.

La proposition précédente n'est pas démontrable, même avec les axiomes d'injectivité et de non-confusion. Afin d'éviter l'introduction d'un axiome supplémentaire, on considère le prédicat  $\text{Nat} : \rightarrow o$  défini par

$$\text{Nat} \equiv n : \forall p : \rightarrow o : [p 0 \Rightarrow \forall x : (p x \Rightarrow p (S x)) \Rightarrow p n] :$$

Intuitivement, la proposition  $\text{Nat } n$  exprime que l'objet  $n$  est dans la plus petite sous-classe du type contenant  $0$  et clos par successeur.

2. Dérivier :  $\text{Nat } 0$

3. Dérivier :  $\forall x : (\text{Nat } x \Rightarrow \text{Nat } (S x))$

Pour travailler dans l'arithmétique, on relativise chaque quantification universelle avec le prédicat  $\text{Nat}$ , et on pos

$$\begin{aligned} \forall x : \text{Nat} : A(x) & \equiv \forall x : (\text{Nat}(x) \Rightarrow A(x)) \\ \exists x : \text{Nat} : A(x) & \equiv \exists x : (\text{Nat}(x) \wedge A(x)) \end{aligned}$$

4. Quels sont les schémas d'introduction et d'élimination associés aux quantificateurs définis ci-dessus ? Dérivier ces schémas.

5. Écrire le principe de récurrence sous forme relativisée, et le démontrer.