

## 2.7.1 — Foundations of Proof Systems

Exam

Nov. 26<sup>th</sup> 2024

*Durée de l'épreuve : 2 heures.* Length of the exam : 2 hours.

### 1 Exercises

#### 1.1 Normal Terms

**Question 1** If  $\alpha, \beta$  are atomic type in simply typed  $\lambda$ -calculus. What are the closed normal terms of respective types :

- $\alpha \rightarrow \alpha$
- $\alpha \rightarrow \alpha \rightarrow \alpha$
- $(\beta \rightarrow \alpha) \rightarrow (\gamma \rightarrow \alpha) \rightarrow \beta \rightarrow \gamma \rightarrow \alpha$   $\diamond$

**Question 2** In System F, what are the possible normal terms  $t$  of these respective types in these respective contexts :

- $[] \vdash t : \forall \alpha . \alpha \rightarrow \alpha$
- $[] \vdash t : \forall \alpha . \alpha \rightarrow \alpha \rightarrow \alpha$
- $[] \vdash t : \forall \alpha . \alpha$
- $[x : \alpha; y : \alpha] \vdash t : \alpha \rightarrow \alpha$   $\diamond$

**Question 3** In Martin-Löf's type theory, what are the possible normal forms for terms  $t$  of these respective types in these respective contexts :

- $[] \vdash t : 4 =_N 4$
- $[] \vdash t : 3 + 2 =_N 5$
- $[] \vdash t : 3 =_N 2$
- $(t \ 4 \ 5)$  where  $[] \vdash t : \Pi a : N. \Pi b : N. \Sigma c : N. (a =_N (\text{add } b \ c)) + (b =_N (\text{add } a \ c))$ .  
(where  $\text{add}$  is the definition of addition).  $\diamond$

## 1.2 Constructivity

**Question 4** Show that, given a propositional variable  $A$ , one can prove :

$$A \Rightarrow \neg\neg A$$

**Question 5** Show that, given a propositional variable  $A$ , one can prove :

$$(A \vee \neg A) \Rightarrow (\neg\neg A \Rightarrow A).$$

(It is ok to describe the proof steps clearly instead of writing the full derivation.)  $\diamond$

**Question 6** Explain why one cannot prove the reverse implication :

$$(\neg\neg A \Rightarrow A) \Rightarrow (A \vee \neg A).$$

## 1.3 Que j'itère

We are in the Calculus of Constructions. We recall the way to code natural numbers :

$$\begin{aligned}\text{nat} &\equiv \Pi X : \text{Type}. X \rightarrow (X \rightarrow X) \rightarrow X \\ 0 &\equiv \lambda X : \text{Type}. \lambda x : X. \lambda f : X \rightarrow X. x \\ S &\equiv \lambda n : \text{nat}. \lambda X : \text{Type}. \lambda x : X. (f (n X x f))\end{aligned}$$

One will write 1, 2, *etc* for  $(S\ 0), (S\ (S\ 0)), \dots$  One also assumes that addition and multiplication have been defined.

One defines :

$$\text{myst} \equiv \lambda n : \text{nat}. \Pi P : \text{nat} \rightarrow \text{Type}. (P\ 1) \rightarrow (\Pi x : \text{nat}. (P\ x) \rightarrow (P\ (\text{mult}\ 2\ x))) \rightarrow (P\ n)$$

**Question 7** What is the type of `myst`?  $\diamond$

**Question 8** Give Proof terms for  $(\text{myst}\ 1)$  and  $(\text{myst}\ 2)$ .  $\diamond$

**Question 9** What is the mathematical “meaning” of  $(\text{myst}\ n)$  (in regular mathematical language)?  $\diamond$

**Question 10** One wants a term

$$\text{div2} : \Pi n : \text{nat}. (\text{myst}\ n) \rightarrow \text{nat}$$

such that  $(\text{div2}\ n\ p)$  reduces to  $\log_2(n)$ . Give such a (as simple as possible) term `div2` (we here mean the original definition of `myst`).  $\diamond$

## 2 Type System with ordinals

In this section we consider an extension of System T (and thus of MLTT) with a form of ordinals.

- One adds a new type  $\text{ord}$  to the system (thus with a rule  $[] \vdash \text{ord} : \text{Type}$ )
- this type has three constructors :
  - $0_o : \text{ord}$
  - $S_o : \text{ord} \rightarrow \text{ord}$
  - $\text{lim} : (N \rightarrow \text{ord}) \rightarrow \text{ord}$  ( $N$  is the usual type of naturals in system T, which comes with its own family of recursors  $R^T$  as studied in the course).

and one recursor over  $\text{ord}$  for any target type  $T$  :

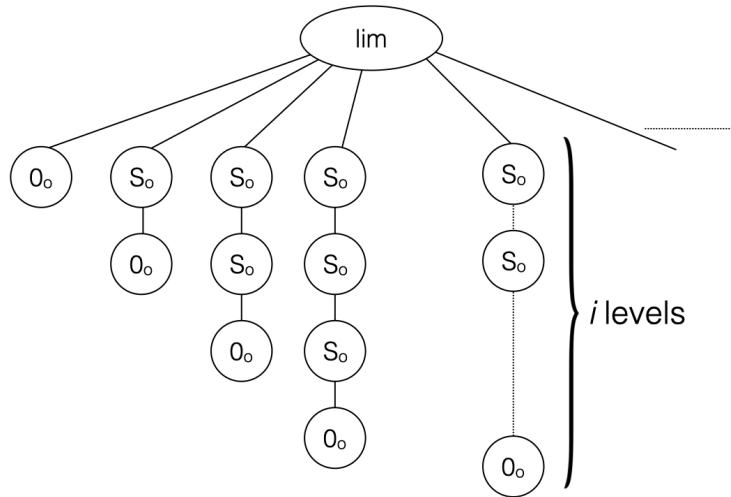
$$\frac{\Gamma \vdash T : \text{Type}}{R^{\text{ord}}_T : T \rightarrow (\text{ord} \rightarrow T \rightarrow T) \rightarrow ((N \rightarrow \text{ord}) \rightarrow (N \rightarrow T) \rightarrow T) \rightarrow \text{ord} \rightarrow T}$$

with the following reduction rules :

1.  $R^{\text{ord}}_T t_0 t_S t_l 0_o \triangleright t_0$
2.  $R^{\text{ord}}_T t_0 t_S t_l (S_o o) \triangleright t_S o (R^{\text{ord}}_T t_0 t_S t_l o)$
3.  $R^{\text{ord}}_T t_0 t_S t_l (\text{lim } f) \triangleright t_l f \lambda n : N. (R^{\text{ord}}_T t_0 t_S t_l (f n))$

**Question 11** Define the straightforward injection  $N\_to\_ord$  of type  $N \rightarrow \text{ord}$ . ◇

Note that the elements of type  $\text{ord}$  can be seen as infinitely branching trees. For instance the element  $(\text{lim } N\_to\_ord)$  can be drawn as :



**Question 12** Define a function  $r : \text{ord} \rightarrow N$  such that, for any closed  $n : N$ , one has  $(r (N\_to\_ord n)) \triangleright n$ . ◇

**Question 13** How can you encode the type  $\text{ord}$  using the impredicative encoding in System F? Give the definitions of  $\text{ord}$ ,  $0_o$  and  $S_o$ . ◇

**Question 14** Give the typing of the  $R^{ord}$  operator(s) in MLTT (that is the version with dependent types).  $\diamond$

**Question 15** In this extension of MLTT, show how to prove

$$\Pi n : N.(r (N\_to\_ord\ n)) =_N n.$$

(The two functions are the ones defined in questions 11 and 12. Please do not go into too much detail; just give the main steps of the proof construction).  $\diamond$

## Strong Normalization

We want to prove strong normalization for this extension of System T (we thus do not need to consider dependent types in this part).

In order to explain the technique, we start by modifying the normalization proof for regular System T, by changing the definition of the reducibility set  $|N|$  (which is  $|T| \equiv SN$  in the “usual” proof).

Like in System F, we define reducibility candidates :

**Définition 1** We call  $SN$  the set of strongly normalizing terms. The set  $\mathcal{N}$  of neutral terms is the set of terms which are not of one of the following forms :  $\lambda x^T.t, 0, (S\ t)$ .  $\diamond$

**Définition 2** A set  $C$  of  $\lambda$ -terms is said to be a reducibility candidate ( $C \in \mathcal{CR}$ ) if and only if these three conditions are verified :

1.  $C \subseteq SN$ ,
2.  $\forall t \in C, t \triangleright_\beta t' \Rightarrow t' \in C$ ,
3.  $\forall t \in \mathcal{N}, (\forall t', t \triangleright t' \Rightarrow t' \in C) \Rightarrow t \in C$ .  $\diamond$

Given two sets of terms  $X$  and  $Y$  one defines :

$$X \rightarrow Y \equiv \{t \mid \forall u \in X, (t\ u) \in Y\}.$$

We recall the two facts which you do not have to show :

- If  $X$  and  $Y$  are reducibility candidates, then  $X \rightarrow Y$  is a reducibility candidate.
- If  $(C_i)_{i \in I}$  is a family of reducibility candidates (resp.  $A \subset \mathcal{CR}$ ) then  $\bigcap_{i \in I} C_i \in \mathcal{CR}$  (resp.  $\bigcap A \in \mathcal{CR}$ ).

In the new proof, we define  $|N|$  as the smallest reducibility candidate such that, for any reducibility candidate  $X$ , we have :

$$(1) \ t \in |N| \Leftrightarrow \forall t_0 \in X, \forall t_S \in |N| \rightarrow X \rightarrow X, (R\ t_0\ t_S\ t) \in X.$$

The two next questions are to show that this set exists; they may be skipped in the first run.

**Question 16** Given sets of terms  $X$  and  $C$  we define

$$F(C, X) \equiv \{t \mid \forall t_0 \in X, \forall t_S \in C \rightarrow X \rightarrow X, (R\ t_0\ t_S\ t) \in X\}.$$

Show that for any  $C$  and  $C'$ ,  $C \subseteq C' \Rightarrow F(C) \subseteq F(C')$ .

Deduce that  $\bigcap_{X \in \mathcal{CR}} F(C, X) \subset \bigcap_{X \in \mathcal{CR}} F(C', X)$   $\diamond$

**Question 17** Show that, if  $C$  and  $X$  are reducibility candidates, then  $F(C, X)$  is a reducibility candidate.

Deduce that, if  $C$  is a reducibility candidate,  $\bigcap_{X \in \mathcal{CR}} F(C, X)$  is a reducibility candidate.  $\diamond$

We then write  $G(C) \equiv \bigcap_{X \in \mathcal{CR}} F(C, X)$ . We can then define :

$$|N| \equiv \bigcap \{C \in \mathcal{CR} \mid G(C) \subseteq G\}$$

which verifies condition (1).

**Question 18** Show that  $0 \in |N|$  and  $S \in |N| \rightarrow |N|$ .  $\diamond$

**Question 19** Show that if, for some type  $T$ ,  $t_0 \in |T|$  and  $t_0 \in |N| \rightarrow |T| \rightarrow |T|$  and  $t \in |N|$ , then  $(R_T t_0 t_S t) \in |T|$ .  $\diamond$

**Question 20** Following the construction above, how can you define the reducibility set  $|\text{ord}|$ ? Roughly describe the steps of the strong normalization proof.  $\diamond$