Institut National
Universitaire
Champollion

Types de données, preuves

Chapitre 3 - Terminaison des fonctions récursives

Révisions
Notion d'ordre bien fondé.

Laura Brillon

Sommaire

Comment prouver la terminaison d'une fonction récursive ?

Ordre bien fondé

Définition

Considérons A un ensemble quelconque, et $<$ un ordre strict sur A. On le note $(A, <)$.

Exemple : L'ensemble des entiers naturels $\mathbb{N}$, muni de son ordre usuel $<$.

Définition

Un ordre strict $<$ sur un ensemble A est **bien fondé**, s'il n'existe pas de suite infinie décroissante.

Exemple : $(\mathbb{N}, <)$ est un ordre bien fondé.

Ordre bien fondé

Exemples

Question : Quel est l'ordre bien fondé ?

a. $(\mathcal{P}(\mathbb{N}), \subset)$
b. $(\mathbb{Z}, <)$
c. $(\mathbb{R}, <)$
d. (Σ^*, "est un préfixe strict de")
e. (Σ^*,ordre lexicographique)

où Σ^* désigne l'ensemble des chaînes de caractères.

Sommaire

$\looparrowright$ À quoi servent les ordres bien fondés ?

Prouver la terminaison

À quoi servent les ordres bien fondés ?

Méthode

Pour prouver qu'une fonction récursive se termine :

- ⋆ Trouver une quantité appartenant à un ensemble A muni d'un ordre bien fondé $<$
- ⋆ Prouver que cette quantité décroit strictement à chaque appel récursif

Remarque : Cette quantité est appelée variant ou convergent est souvent un des paramètres d'appel de la fonction.

Terminaison d'une fonction récursive

Prouver la terminaison

Exemple 1 : Prouver la terminaison de la fonction suivante :

```
let rec facto = function
0 -> 1
| n -> n * facto(n-1) ;;
```

Exemple 2 : Comment prouver la terminaison de la fonction `positif` ?

```
let rec positif = function
 [ ] -> 0
| x :: q -> if x < 0 then positif(q)
else 1 + positif(q) ;;
```

Sommaire

À ce stade, nous manquons d'ordres bien fondés !

↬ **Comment construire de nouveaux ordres bien fondés ?**

Exemples d'ordres bien fondés

Mesure

Point de départ :

- A un ensemble muni d'un ordre bien fondé $<_A$
- B un autre ensemble

Objectif : Nous souhaitons munir B d'un ordre bien fondé.

Ingrédient nº 1 : Trouvons une fonction $m : B \to A$.
La fonction m est appelée une *mesure*.

Ingrédient nº 2 : Définissons une relation $<_B$ sur l'ensemble B par

$$x <_B y \iff m(x) <_A m(y)$$

Proposition

Alors $<_B$ est un ordre bien fondé sur B.

Mesure - Exemple

Question

Question : Pour prouver la terminaison de la fonction `positif` ci-contre, quelle mesure utiliseriez-vous ?

a. $m : \begin{cases} \mathbb{N} \to \{liste\} \\ n \mapsto \{1; 2; \ldots; n\} \end{cases}$

b. $m : \begin{cases} \mathbb{N} \to \{liste\} \\ n \mapsto longueur(n) \end{cases}$

c. $m : \begin{cases} \{liste\} \to \mathbb{N} \\ \ell \mapsto longueur(\ell) \end{cases}$

d. $m : \begin{cases} \{liste\} \to \mathbb{N} \\ \ell \mapsto card(positif(\ell)) \end{cases}$

```
let rec positif = function
 [ ] -> 0
| x :: q -> if x < 0 then
positif(q)
else 1 + positif(q) ;;
```

Terminer la preuve de la terminaison de cette fonction.

Sommaire

Exemples d'ordres bien fondés

Ordre lexicographique

Remarque - Caml connaît cet ordre lexicographique !

Question : Qui est l'intrus ?

a. $(3, 4) < (1, 5)$

b. $(3, \textit{true}) > (3, \textit{false})$

c. $(3, 2, 1) > (3, 2, 0)$

d. $(3,' a') < (3,' b')$

Exemples d'ordres bien fondés

Ordre lexicographique

Définition

Soient $(A, >_A)$ et $(B, >_B)$ deux ensembles munis d'un ordre.
L'ordre lexicographique $>_{lex}$ est défini sur $A \times B$ par :

$$(a, b) >_{lex} (a', b') \iff (a >_A a') \text{ ou } ((a = a') \text{ et } (b >_B b'))$$

Remarque - Peut se généraliser pour n'importe quel produit cartésien $A_1 \times A_2 \times \ldots \times A_n$.

Attention - Ne pas confondre avec l'ordre lexicographique sur les chaînes de caractères !

Exemples d'ordres bien fondés

Ordre lexicographique

Proposition

Si $>_A$ et $>_B$ sont bien fondés,

alors $>_{lex}$ est un ordre bien fondé.

Les idées de la preuve

Idées de la preuve.

- Raisonnement par l'absurde : Supposons que $>_{lex}$ n'est pas bien fondé.
- On traduit ce que cela signifie : Il existe une suite infinie décroissante

$$(S) : (a_1, b_1) > (a_2, b_2) > \ldots > (a_n, b_n) > (a_{n+1}, b_{n+1}) > \ldots$$

- $(A, >_A)$ et $(B, >_B)$ sont bien fondés. On aboutit à une contradiction en montrant que la suite infinie (S) conduit à une suite infinie décroissante sur A ou sur B.

$\square$

Exemples d'ordres bien fondés

Ordre lexicographique

Application : Terminaison d'une fonction

Exemple 1

Justifier la terminaison de cette fonction :

```
let rec f (x,y) =
if x=0 || y=0 then 0
else if x=1 then f(x,y-1)
else f(x-1,y) ;;


- : int * int -> int = <fun>
```

Exemples d'ordres bien fondés

Ordre lexicographique

Exemple 2 - Fonction d'Achermann

L'exemple le plus célèbre est la fonction d'Ackermann

```
 let rec ack = function
(0,y) -> y+1
| (x,0) -> ack(x-1 , 1)
| (x,y) -> ack(x-1, ack(x,y-1)) ;;
```

Pause Culture : Définie en 1928 par W. Ackermann comme exemple de fonction

- qui termine - en faire la preuve !
- récursive non récursive primitive (Théorie des langages)

Sommaire

Lié au Chapitre suivant, sur le problème d'Unification

Exemples d'ordres bien fondés

Multiensembles

Les multiensembles sont des "ensembles" dans lesquels on s'autorise des répétitions.

Exemple

- $\{true,\ true,\ false,\ false,\ false\}$
- $\{0, 1, 1, 2, 3, 3, 3\}$

L'ordre des éléments dans un multiensemble n'a pas d'importance.

Exemples d'ordres bien fondés

Multiensembles

Nous avons vu (TP n° 1) qu'il est possible de définir un ordre sur les multi-ensembles **finis** d'entiers naturels :

Définition

Si M et N sont deux multi-ensembles tels que $M \neq N$,

- on calcule l'intersection de M et N : $I = M \cap_m N$.
- on détermine $X = M \setminus_m I$. Il faut que $X \neq \emptyset$.
- on détermine $Y = N \setminus_m I$.
- on regarde si l'élément maximal de X est plus grand que tous les éléments de Y.

Dans ce cas, on dit que $N < M$.

Exemples d'ordres bien fondés

Multiensembles

Exemple

$\{5, 3, 2\} > \{5, 2, 2, 1\}$
En effet, $X = \{3\}$, $Y = \{2, 1\}$.

Question : Qui est l'intrus ?

1. $\{5, 3, 2\} > \{4, 4, 4, 2, 2\}$
2. $\{5, 3, 2\} > \{5, 5\}$
3. $\{5, 3, 2\} > \{5, 1, 1, 1\}$
4. $\{5, 3, 2\} > \{5, 3\}$

Multiensembles

Proposition

Cet ordre sur les multi-ensembles d'entiers naturels (noté $>_{mul}$) est total et **bien fondé**

Application : Dans le prochain chapitre, nous verrons l'algorithme d'Unification (qui sert à l'Inférence de types) et nous prouverons la terminaison de cet algorithme grâce à cet ordre sur les multiensembles.