TD8: Induction structurelle
Correction

MP2I Lycée Pierre de Fermat

Exercice 1. *Entiers de Péano*

Q1. On définit inductivement la fonction **int** comme suit :

- $\mathbf{int}(Z) = 0$
- Pour $a \in \mathcal{P}$, $\mathbf{int}(S(a)) = 1 + \mathbf{int}(a)$

Q2. Montrons par induction sur la structure de $\mathcal{P}$ la propriété suivante :

$$\forall a \in \mathcal{P}, I(a) : \text{“}\forall b \in \mathcal{P}, \mathbf{int}(\mathbf{add}(a,b)) = \mathbf{int}(a) + \mathbf{int}(b)\text{”}$$

Il y a deux cas :

- $a = Z$: Soit $b \in \mathcal{P}$. $\mathbf{int}(\mathbf{add}(Z,b)) = \mathbf{int}(b)$ par définition de **add**, et $\mathbf{int}(Z) + \mathbf{int}(b) = \mathbf{int}(b)$ par définition de **int**. D'où $I(Z)$.
- $a = S(a')$ avec $a' \in \mathcal{P}$. Hypothèse d'induction : $I(a')$, i.e. $\forall b \in \mathcal{P}, \mathbf{int}(\mathbf{add}(a',b)) = \mathbf{int}(a) + \mathbf{int}(b)$. On a (**Note : pour cette question je mets toutes les justifications pour vous expliquer chaque passage de ligne, mais dans une vraie copie ce n'est pas la peine, en revanche il faut systématiquement le faire pour les hypothèses d'induction et pour l'utilisation de résultats des questions précédents**) :

$$\begin{aligned}
\mathbf{int}(\mathbf{add}(a,b)) &= \mathbf{int}(\mathbf{add}(S(a'),b)) \\
&= \mathbf{int}(S(\mathbf{add}(a',b))) && \text{par définition de } \mathbf{add} \\
&= 1 + \mathbf{int}(\mathbf{add}(a',b)) && \text{par définition de } \mathbf{int} \\
&= 1 + \mathbf{int}(a') + \mathbf{int}(b) && \text{par hypothèse d'induction} \\
&= \mathbf{int}(a) + \mathbf{int}(b) && \text{par définition de } \mathbf{int}
\end{aligned}$$

D'où $I(a)$.

Q3. On raisonne par induction structurelle sur $a \in \mathcal{P}$. Il y a deux cas :

- $a = Z$: $\mathbf{add}(a,Z) = \mathbf{add}(Z,Z) = Z = a$
- $a = S(a')$ avec $a' \in \mathcal{P}$. Hypothèse d'induction : $\mathbf{add}(a',Z) = a'$. On a :

$$\begin{aligned}
\mathbf{add}(a,Z) &= \mathbf{add}(S(a'),Z) \\
&= S(\mathbf{add}(a',Z)) \\
&= S(a') && \text{par hypothèse d'induction} \\
&= a
\end{aligned}$$

Q4. On raisonne par induction structurelle sur $a \in \mathcal{P}$. Il y a deux cas :

— $a = Z$:

$$\begin{aligned}\mathbf{add}(S(Z), b) &= S(\mathbf{add}(Z, b)) \\ &= S(b) \\ &= \mathbf{add}(Z, S(b))\end{aligned}$$

— $a = S(a')$ avec $a' \in \mathcal{P}$. Hypothèse d'induction : $\forall b \in \mathcal{P}, \mathbf{add}(S(a'), b) = S(\mathbf{add}(a', b))$. Soit $b \in \mathcal{P}$. On a :

$$\begin{aligned}\mathbf{add}(S(a), b) &= \mathbf{add}(S(S(a')), b) \\ &= S(\mathbf{add}(S(a'), b)) \\ &= S(\mathbf{add}(a', S(b)) \quad \text{par hypothèse d'induction} \\ &= \mathbf{add}(S(a'), S(b)) \\ &= \mathbf{add}(a, S(b))\end{aligned}$$

Q5. Montrons par induction structurelle sur $a \in \mathcal{P}$ la propriété suivante :

$$\forall a \in \mathcal{P}, \mathbf{Com}(a) : \text{“}\forall b \in \mathcal{P}, \mathbf{add}(a, b) = \mathbf{add}(b, a)\text{”}$$

Il y a deux cas :

— $a = Z$. Alors, pour $b \in \mathcal{P}$, $\mathbf{add}(a, b) = \mathbf{add}(Z, b) = b = \mathbf{add}(b, Z)$ d'après la question 3. Donc $\mathbf{Comm}(Z)$.

— $a = S(a')$ avec $a' \in \mathcal{P}$. HI : $\mathbf{Comm}(a')$, i.e. $\forall b \in \mathcal{P}, \mathbf{add}(a', b) = \mathbf{add}(b, a')$. Pour $b \in \mathcal{P}$, on a :

$$\begin{aligned}\mathbf{add}(a, b) &= \mathbf{add}(S(a'), b) \\ &= S(\mathbf{add}(a', b) \\ &= S(\mathbf{add}(b, a') \qquad \text{par HI} \\ &= \mathbf{add}(S(b), a') \\ &= \mathbf{add}(b, S(a')) \quad \text{d'après la question 4} \\ &= \mathbf{add}(b, a)\end{aligned}$$

D'où $\mathbf{Com}(a)$.

Ainsi, par principe d'induction structurelle, $\forall a, b \in \mathcal{P}, \mathbf{add}(a, b) = \mathbf{add}(b, a)$.

Q6. Montrons que $\forall a, b, c \in \mathcal{P}, \mathbf{add}(a, \mathbf{add}(b, c)) = \mathbf{add}(\mathbf{add}(a, b), c)$. Fixons $b, c \in \mathcal{P}$, et montrons par induction structurelle sur $a \in \mathcal{P}$ que :

$$\forall a \in \mathcal{P}, \mathbf{Assoc}(a) : \text{“}\mathbf{add}(a, \mathbf{add}(b, c)) = \mathbf{add}(\mathbf{add}(a, b), c)\text{”}$$

Il y a deux cas :

— $a = Z$. On a :

$$\begin{aligned}\mathbf{add}(a, \mathbf{add}(b, c)) &= \mathbf{add}(Z, \mathbf{add}(b, c)) \\ &= \mathbf{add}(b, c) \\ &= \mathbf{add}(\mathbf{add}(Z, b), c) \\ &= \mathbf{add}(\mathbf{add}(a, b), c)\end{aligned}$$

— $a = S(a')$ avec $a' \in \mathcal{P}$. Hypothèse d'induction : $\mathbf{add}(a', \mathbf{add}(b, c)) = \mathbf{add}(\mathbf{add}(a', b), c)$. On a :

$$
\begin{aligned}
\mathbf{add}(a, \mathbf{add}(b, c)) &= \mathbf{add}(S(a'), \mathbf{add}(b, c)) \\
&= S(\mathbf{add}(a', \mathbf{add}(b, c)) \\
&= S(\mathbf{add}(\mathbf{add}(a', b), c) \quad \text{par HI} \\
&= \mathbf{add}(S(\mathbf{add}(a', b)), c) \\
&= \mathbf{add}(\mathbf{add}(S(a'), b), c) \\
&= \mathbf{add}(\mathbf{add}(a, b), c)
\end{aligned}
$$

Q7. Il y a plusieurs manières de définir la multiplication, chacune menant à une preuve plus ou moins compliquées et nécessitant plus ou moins de lemmes intermédiaires. Une astuce est de déconstruire les deux côtés de la multiplication en même temps, en utilisant le fait que $(n+1) \times (m+1) = n \times m + n + m + 1$. Ainsi, on définit inductivement la multiplication comme suit :

— Pour tout $b \in \mathcal{P}, \mathbf{mult}(Z, b) = Z$
— Pour tout $a \in \mathcal{P}, \mathbf{mult}(a, Z) = Z$
— Pour tout $a, b \in \mathcal{P}, \mathbf{mult}(S(a), S(b)) = S(\mathbf{add}(\mathbf{mult}(a, b), \mathbf{add}(a, b))$

On peut alors montrer la commutativité et la distributivité sans lemme, en utilisant directement les propriétés de l'addition. Par exemple pour la commutativité, montrons par double induction sur $a, b \in \mathcal{P}$ que $\forall a, b \in \mathcal{P}, \mathbf{mult}(a, b) = \mathbf{mult}(b, a)$. Il y a trois cas :

— $a = Z$. $\mathbf{mult}(Z, b) = Z = \mathbf{mult}(b, Z)$.
— $b = Z$: analogue.
— $a = S(a'), b = S(b')$ avec $a', b' \in \mathcal{P}$.

$$
\begin{aligned}
\mathbf{mult}(a, b) &= \mathbf{mult}(S(a'), S(b')) \\
&= S(\mathbf{add}(\mathbf{mult}(a', b'), \mathbf{add}(a', b')) \\
&= S(\mathbf{add}(\mathbf{mult}(b', a'), \mathbf{add}(a', b')) && \text{par HI, car } (a', b') < (a, b) \\
&= S(\mathbf{add}(\mathbf{mult}(b', a'), \mathbf{add}(b', a')) && \text{par commutativité de l'addition} \\
&= \mathbf{mult}(S(b'), S(a')) && \text{par définition de } \mathbf{mult} \\
&= \mathbf{mult}(b, a)
\end{aligned}
$$

La preuve d'associativité est similaire.

On pouvait aussi définir la multiplication comme suit (plus proche de ce qui a été fait pour l'addition) en traduisant que $(n+1) \times m = n \times m + m$:

— $\mathbf{mult}(Z, b) = Z$
— $\mathbf{mult}(S(a), b) = \mathbf{add}(\mathbf{mult}(a, b), b)$

Dans ce cas, la preuve nécessite plusieurs lemmes, notamment la distributivité de la multiplication sur l'addition, mais tout se fait par induction simple sur a sans piège.

Exercice 2.

Entiers de Péano (partie 2)

Exercice 3. *Expressions*

Q1. On définit inductivement la fonction n_{op} :

- Pour $n \in \mathbb{N}$, $n_{op}(\mathbf{Int}(n)) = 0$
- Pour $x \in \mathcal{X}, n_{op}(x) = 0$
- Pour $e_1, e_2 \in \mathcal{E}, n_{op}(e_1 + e_2) = 1 + n_{op}(e_1) + n_{op}(e_2)$
- Pour $e_1, e_2 \in \mathcal{E}, n_{op}(e_1 \times e_2) = 1 + n_{op}(e_1) + n_{op}(e_2)$

Q2. On définit de même la fonction n_{atom} :

- Pour $n \in \mathbb{N}$, $n_{atom}(\mathbf{Int}(n)) = 1$
- Pour $x \in \mathcal{X}, n_{atom}(x) = 1$
- Pour $e_1, e_2 \in \mathcal{E}, n_{atom}(e_1 + e_2) = n_{atom}(e_1) + n_{atom}(e_2)$
- Pour $e_1, e_2 \in \mathcal{E}, n_{atom}(e_1 \times e_2) = n_{atom}(e_1) + n_{atom}(e_2)$

Q3. Montrons par induction sur la structure des expressions que pour $e \in \mathcal{E}, n_{\text{op}}(e) + 1 = n_{\text{atom}}(e)$.
Il y a 4 cas :

- Si $e = \mathbf{Int}(n)$ avec $n \in \mathbb{N}$, $n_{op}(e) = 0$ et $n_{atom}(e) = 1$: la propriété est vraie
- Si $e = x \in \mathcal{X}$, $n_{op}(e) = 0$ et $n_{atom}(e) = 1$: la propriété est vraie
- Si $e = e_1 + e_2$ avec $e_1, e_2 \in \mathcal{E}$: par hypothèse d'induction, on a $n_{\text{op}}(e_1) + 1 = n_{\text{atom}}(e_1)$ et $n_{\text{op}}(e_2) + 1 = n_{\text{atom}}(e_2)$. Donc :

$$n_{atom}(e) = n_{atom}(e_1) + n_{atom}(e_2) = n_{\text{op}}(e_1) + 1 + n_{\text{op}}(e_2) + 1 = n_{op}(e) + 1$$

- Si $e = e_1 \times e_2$ avec $e_1, e_2 \in \mathcal{E}$, c'est analogue.

Q4. Définissons inductivement une fonction $\mathbf{eval}(e, \sigma)$ qui, étant donné une expression e et un contexte σ, calcule la valeur de e avec le contexte σ :

- $\mathbf{eval}(\mathbf{Int}(n), \sigma) = n$
- $\mathbf{eval}(x, \sigma) = \sigma(x)$
- $\mathbf{eval}(e_1 + e_2, \sigma) = \mathbf{eval}(e_1, \sigma) + \mathbf{eval}(e_2, \sigma)$
- $\mathbf{eval}(e_1 \times e_2, \sigma) = \mathbf{eval}(e_1, \sigma) \times \mathbf{eval}(e_2, \sigma)$

Il faut bien faire attention ici que le signe + a deux sens : dans le monde des expressions $\mathcal{E}$, c'est juste une notation syntaxique, et dans le monde des entiers $\mathbb{N}$, c'est l'addition standard. Par exemple, $e_1 + (e_2 + e_3)$ et $(e_1 + e_2) + e_3$ sont deux expressions différentes, mais lorsqu'on les évalue, on obtient le même résultat.

Exercice 4.

Récursivité terminale

Q1. Notons C_n le coût de la fonction sur une liste de taille n. On a $C_n = C_{n-1} + \mathcal{O}(n)$ car la concaténation est linéaire en la taille de la liste de gauche. Ainsi, $C_n = \mathcal{O}(n^2)$.

Q2. On reprend la version vue en TP avec un accumulateur :

```
1 (* renvoie (rev l) @ acc *)
2 let rec rev_concat (l: 'a list) (acc: 'a list) : 'a list =
3   match l with
4   | [] -> acc
5   | x :: q -> rev_concat l (x :: acc)
6
7 (* Renvoie une copie de l dans l'ordre inverse *)
8 let rev_version2 (l: 'a list) : 'a list =
9   rev_concat l []
```

Fixons E un ensemble, notons $\mathcal{L}$ l'ensemble des listes sur E, et montrons par induction structurelle sur les listes que $\forall l \in \mathcal{L}, P(L)$: "$\forall A \in \mathcal{L}, \textbf{rev\_concat}(L, A) = \textbf{rev}(L)@A$".

Il y a a deux cas :

— $L = []$: Soit $A \in \mathcal{L}$. Par définition, $\textbf{rev\_concat}([], A) = A = \textbf{rev}([])@A$, d'où $P([])$.

— $L = x :: Q$ avec $x \in E, Q \in \mathcal{L}$. Soit $A \in \mathcal{L}$. Alors :

$$\begin{aligned} \textbf{rev\_concat}(L, A) &= \textbf{rev\_concat}(Q, x :: A) \\ &= \textbf{rev}(Q)@x :: A && \text{par HI} \\ &= \textbf{rev}(Q)@[x]@A \\ &= \textbf{rev}(L)@A && \text{par définition de } \textbf{rev} \end{aligned}$$

D'où $P(L)$.

Ainsi, $\forall L \in \mathcal{L}, P(L)$.

En particulier, en prenant $A = []$, on a que pour toute liste L, $\textbf{rev\_concat}(L, []) = \textbf{rev}(L)$. Ceci montre bien que la fonction **rev_version2** est identique à **rev**.

Q3. On procède ici par induction bien fondée. Notons $P(L_1, L_2, L_3)$ la propriété à démontrer. Soit (L_1, L_2, L_3) trois listes quelconques, telles que pour tout triplet $(L_1', L_2', L_3') < (L_1, L_2, L_3)$, on a $P(L_1', L_2', L_3')$.

— Si $L_1 = []$, alors $\textbf{rfc}([], L_2, L_3) = \textbf{rev}(L_2)@L_3 = \textbf{rev}(\textbf{fusion}(L_1, L_2))@L_3$. D'où $P(L_1, L_2, L_3)$.

— Si $L_2 = []$, analogue.

— Si $L_1 = x_1 :: Q_1$ et $L_2 = x_2 :: Q_2$ avec $x_1, x_2 \in E$ et $Q_1, Q_2 \in L$: Supposons sans perte de généralité que $x_1 < x_2$, l'autre cas étant identique. Alors :

$$\begin{aligned} \textbf{rfc}(L_1, L_2, L_3) &= \textbf{rfc}(Q_1, L_2, x_1 :: L_3) \\ &= \textbf{rev}(\textbf{fusion}(Q_1, L_2))@x_1 :: L_3 && \text{par HI} \\ &= \textbf{rev}(\textbf{fusion}(Q_1, L_2))@[x_1]@L_3 \\ &= \textbf{rev}(x_1 :: \textbf{fusion}(Q_1, L_2))@L_3 && \text{par définition de } \textbf{rev} \\ &= \textbf{rev}(\textbf{fusion}(x_1 :: Q_1, L_2))@L_3 && \text{par définition de } \textbf{fusion} \\ &= \textbf{rev}(\textbf{fusion}(L_1, L_2))@L_3 \end{aligned}$$

D'où $P(L_1, L_2, L_3)$.

Ainsi, par principe d'induction bien fondée, la propriété annoncée est bien vraie pour toutes listes (L_1, L_2, L_3).

Q4. En particulier, pour $L_3 = []$, on a que pour toutes listes L_1, L_2 :

$$\begin{aligned} \mathbf{fusion2}(L_1, L_2) &= \mathbf{rev}(\mathbf{rfc}(L_1, L_2, [])) \\ &= \mathbf{rev}(\mathbf{rev}(\mathbf{fusion}(L_1, L_2)@[]) \\ &= \mathbf{rev}(\mathbf{rev}(\mathbf{fusion}(L_1, L_2)) \\ &= \mathbf{fusion}(L_1, L_2) \end{aligned}$$

Exercice 5.

Preuves sur les listes

Exercice 6.

Induction bien fondée

Remarquons que les appels récursifs à f se font sur des entrées strictement décroissantes dans l'ordre lexicographique.

Montrons par induction bien fondée sur $\mathbb{N}^2$ muni de l'ordre lexicographique que :

$$\forall (x, y) \in \mathbb{N}^2, P(x, y) : \text{“}f(x, y) = \frac{x(x+3)}{2} + y\text{”}$$

Soit $(x, y) \in \mathbb{N}^2$ tel que pour tout couple $(x', y') < (x, y)$, $P(x', y')$ est vérifiée.

— Si $x = 0$ et $y = 0$, $f(x, y) = 0$, et $P(0, 0)$ est bien vérifiée.

— Sinon, si $x > 0$ et $y = 0$, alors $f(x, 0) = f(x-1, x+1)$. Par HI, cela vaut $\frac{(x-1)(x+2)}{2} + x + 1 = \frac{x(x+3)}{2}$, d'où $P(x, y)$.

— Sinon, $f(x, y) = f(x, y-1) + 1$. Par HI, cela vaut $(\frac{x(x+3)}{2} + y - 1) + 1$, soit exactement $\frac{x(x+3)}{2} + y$: $P(x, y)$ est vraie.

Ainsi, par principe d'induction bien fondée, $\forall (x, y) \in \mathbb{N}^2, f(x, y) = \frac{x(x+3)}{2} + y$.

Exercice 7.

Mi casa es tu casa

Exercice 8.

Le retour de Knaster-Tarski

Q1. Montrons que toute fonction continue est croissante. Soit $f : (X, \leq_X) \to (Y, \leq_Y)$ continue, avec X, Y des treillis complets. Soient $x_1, x_2 \in X$ avec $x_1 \leq x_2$. On considère $A = \{x_1, x_2\}$. A est non vide et totalement ordonnée, donc $f(\bigvee A) = \bigvee(f(A))$. De plus, $\bigvee(A) = x_2$. Donc, $f(x_2) = \bigvee(\{f(x_1), f(x_2)\})$, et donc $f(x_2) \geq f(x_1)$. La fonction f est bien croissante.

Q2. $\bot$ est le plus petit élément de X. On pose $p_0 = \bot$, et $p_{n+1} = f(p_n)$ pour $n \in \mathbb{N}$. Montrons que la suite $(p_n)_n$ est croissante. $p_0 \leq p_1$ car $p_0 = \bot$. Donc, pour $n \in \mathbb{N}$, en appliquant f n fois, on obtient par croissance de f que $f^n(p_0) \leq f^{n+1}(p_1)$, i.e. que $p_n \leq p_{n+1}$. La suite est bien croissante.

Q3. Montrons que $p = \bigvee\{p_n | n \in \mathbb{N}\}$ est un point fixe de f. Par continuité de f, $f(p) = f(\bigvee\{p_n | n \in \mathbb{N}\}) = \bigvee\{f(p_n) | n \in \mathbb{N}\}$. Donc, $f(p) = \bigvee\{p_{n+1} | n \in \mathbb{N}\}$, et comme $(p_n)_n$ est croissante, cela donne bien $f(p) = \bigvee\{p_n | n \in \mathbb{N}\} = p$.

Montrons maintenant que c'est le plus petit point fixe. Soit p' un autre point fixe de f. $p' \geq \bot = p_0$, donc par croissance de f, en itérant f sur l'inégalité, on obtient que $f^n(p') \geq p_n$ pour $n \in \mathbb{N}$. Donc, $p' \geq p_n$ pour $n \in \mathbb{N}$. Donc p' est un majorant de $\{p_n | n \in \mathbb{N}\}$, donc $p' \geq p$.

Q4. $\Phi(\emptyset)$ vaut $\emptyset \cup B \cup \emptyset = B$: c'est l'ensemble des termes construits avec les règles de base. $\Phi(\Phi(\emptyset))$ contient B ainsi que l'ensemble des termes construits à partir des règles d'induction sur les éléments de B, autrement dit les termes de hauteur 1, autrement dit ce que l'on a noté X_1 dans le cours. De manière générale, $\Phi^n(\emptyset)$ représente les termes inductifs de hauteur au plus $n-1$ pour $n \in \mathbb{N}^*$.

Q5. Commençons par montrer la croissante de Φ. Si $A \subseteq A'$ sont deux parties de X, alors $f(A) \subseteq f(A')$. En effet :

- $A \subseteq A'$
- $B \subseteq B$
- en notant $I_A = \{S(p, x_1, \ldots, x_r) \mid (S, r, P) \in \mathcal{I}, x_1, \ldots, x_r \in A, p \in P\}$, on a bien $I_A \subseteq I'_A$: si $S(p, x_1, \ldots, x_r) \in I_A$, alors $x_1, \ldots, x_r \in A$ et donc $x_1, \ldots x_r \in A'$, et donc $S(p, x_1, \ldots, x_r) \in I'_A$).

Continuité : pas beaucoup plus compliqué, il faut juste dérouler les définition.

Q6. Le plus petit point fixe de Φ est donc la limite des $\Phi^n(\emptyset)$, autrement dit c'est $\bigcup_{n\in\mathbb{N}} X_n$, c'est à dire X tout entier.

Q7. Montrons que $\Phi(Q) = Q$. Cela permettra de montrer que Q est un point fixe, et donc que $Q = X$ d'après la question précédente.

- $f(Q)$ contient Q par définition de Φ.
- Q contient $\Phi(Q)$ car Q contient B, Q et I_Q : en effet si $x = S(p, x_1, \ldots, x_r) \in I_Q$ avec $x_1, \ldots, x_r \in Q$, alors $x_1, \ldots, x_r$ vérifient $\mathcal{Q}$, et donc x aussi par hypothèse sur $\mathcal{Q}$.

D'où l'égalité.
Donc, $Q = X$, autrement dit tous les éléments de X vérifient Q.

Exercice 9.

Existence des fonctions inductives