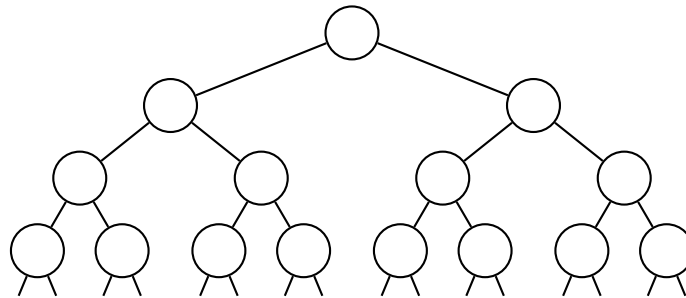


TD n°15 - Corrigé

Exercice 1 Arbres binaires parfaits

On considère des arbres non-étiquetés uniquement.

1. Dessiner un arbre binaire parfait de hauteur 3.



2. Dans un arbre parfait, que peut-on remarquer sur les sous-arbres gauches et droits de la racine ?

Ce sont des arbres parfaits de même hauteur. Puisqu'on considère des arbres non étiquetés, ce sont les mêmes arbres.

3. En déduire une définition récursive pour l'ensemble des arbres binaires parfaits. (seulement 1 élément de base et 1 constructeur sont nécessaires)

Vide est un arbre parfait.

$$\mathcal{B} = \text{Vide}$$

$$C = \{g : a \mapsto N(a, a)\}$$

4. Prouver par induction structurelle que pour tout arbre parfait a , on a $n(a) = 2^{h(a)+1} - 1$ avec $n(a)$ le nombre de sommets de l'arbre et $h(a)$ sa hauteur.

Raisonnons selon la structure de a en tant qu'arbre binaire parfait (avec la définition inductive précédente)

- Si $a = \text{Vide}$ alors $n(a) = 0$ et $h(a) = -1$, donc $n(a) = 2^{h(a)+1} - 1$
- Si $a = g(b)$ en supposant $n(b) = 2^{h(b)+1} - 1$. On a $n(a) = 1 + 2 * n(b)$ et $h(a) = h(b) + 1$.
Donc $n(a) = 1 + 2 * n(b) = 1 + 2 * (2^{h(b)+1} - 1) = 2^{h(b)+2} - 1 = 2^{h(a)+1} - 1$.

Exercice 2 Un ensemble de mots

On définit Σ l'ensemble des lettres de l'alphabet latin.

On considère l'ensemble $\mathcal{T}$ défini récursivement par $\mathcal{B} = \Sigma$ et $C = \{f_\alpha : m \mapsto \alpha m \alpha \mid \forall \alpha \in \Sigma\}$ (ce qui veut dire qu'on a 26 constructeurs). Dans cet exemple, les termes construits sont des mots.

1. Les mots suivants appartiennent-ils à $\mathcal{T}$? Si oui, comment peut-on les définir avec les éléments de la base et les constructeurs.

- $a \in \mathcal{B}$
- dce non
- $dcd = f_d(c)$
- $bbbb$ non
- $bbebb = f_b(f_b(e))$

2. Montrer par induction structurelle que tous les mots dans $\mathcal{T}$ sont de longueur impaire.

Raisonnons selon la structure d'un terme $t \in \mathcal{T}$:

- si $t \in \mathcal{B}$, t est un mot de longueur 1, impaire.
- si t s'écrit $f_\alpha(m)$ pour α une lettre et m un autre terme, en supposant que m est de longueur impaire, alors t s'écrit $\alpha m \alpha$ et a comme longueur $|t| = |m| + 2$. Comme $|m|$ est impaire, $|t|$ aussi.

Par principe d'induction, tout $t \in \mathcal{T}$ est de longueur impaire.

3. Montrer également que tout mot de $\mathcal{T}$ est un palindrome. Pour rappel, une chaîne de caractères $s_0 s_1 \dots s_n$ est un palindrome si et seulement si pour tout $i \in [0, n]$, $s_i = s_{n-i}$

Raisonnons selon la structure d'un terme $t \in \mathcal{T}$:

- si $t \in \mathcal{B}$, t est un palindrome a une lettre.
 - si t s'écrit $f_a(m)$ pour a un lettre et m un autre terme, en supposant que m est un palindrome, alors t s'écrit ama .
- Notons n la longueur de t . Par hypothèse d'induction tout $i \in [1, n-1]$, $t_i = t_{n-i}$. De plus $t_0 = t_n = a$.
Donc t est un palindrome.

Par principe d'induction, tout $t \in \mathcal{T}$ est un palindrome.

Exercice 3 Mots bien parenthésés

1. ab oui : $\text{Concat}(a, b)$, (a) oui : $\text{Par}(a)$, $(a \text{ non : mal parenthésé, } (a(b)) \text{ oui : } \text{Par}(\text{Concat}(a, \text{Par}(b)))$, $((bb(b)b(bb)bb)))$ non : mal parenthésé.
2. a peut être défini comme $\text{Concat}(\epsilon, a)$ ou juste a .
3. Montrer que tout mot bien parenthésé u vérifie 1. $p(u) = 0$ et 2. si v est un préfixe de u , alors $p(v) \geq 0$.
On le montre par induction structurelle sur u .

- Si $u = \epsilon$ alors $p(u) = 0$. et u n'a pas d'autre préfixe que ϵ
- Si $u = a$, $a \in \Sigma$, $a \neq ($ et $a \neq)$, alors $p(u) = 0$. u n'a pas d'autre préfixe que ϵ et u .
- Si $u = \text{Par}(v)$ avec v qui vérifie l'hypothèse d'induction, alors $u = (v)$. On a alors $p(u) = p(() + p(v) + p()) = 1 + -1 = 0$. Les préfixes de u sont u , $($ qui vérifie $p(() \geq 0$, $(v'$ avec v' un préfixe de v , qui vérifie $p((v') = 1 + p(v') \geq 0$ et enfin (v) qui vérifie $p(u) = 0$.
- Si $u = \text{Concat}(v, w)$ avec v et w qui vérifient l'hypothèse de récurrence alors $p(u) = p(v) + p(w) = 0$. Les préfixes de u sont soit des préfixes de v , qui ont un poids positif par hypothèse d'induction, soit vw' avec w' un préfixe de w . le poids est alors $p(v) + p(w') = p(w') \geq 0$.

4. Montrer réciproquement que si $u \in \Sigma^*$ satisfait ces propriétés, alors u est dans $\mathcal{T}$.

Ici on n'a pas de structure inductive sur laquelle faire une induction : u n'est pas supposé élément de $\mathcal{T}$, **c'est ce qu'on veut montrer**.

On va donc raisonner par récurrence forte sur le nombre de lettres (sa taille) du mot u (raisonnement classique, vous reverrez ça l'an prochain).

- Si u est de taille 0, $u = \epsilon$ donc $u \in \mathcal{T}$.
- Si u est de taille 1, $u = a$ pour une certaine lettre a qui n'est ni $($ ni $)$ puisque $p(u) = 0$. Donc $u \in \mathcal{T}$.
- Supposons la propriété vraie pour tout $k \leq n$. Soit u de taille $n + 1$. Raisonnons selon la première lettre u_0 :
 - Ça ne peut pas être $)$, sinon $v = u_0$, qui est un préfixe de u , vérifie $p(v) \leq 0$.
 - Si c'est $($, alors on considère le plus petit i tel que $p(u_0 \dots u_i) = 0$ (logiquement $u_i =)$ mais on a pas besoin de le montrer).
Si $i = n$, alors u s'écrit (v) avec v de longueur inférieure à n qui vérifie $p(v) = p(u) + 1 - 1 = 0$. De plus pour tout préfixe v' de v , $(v'$ est un préfixe de u , donc $p((v') \geq 0$ ce qui entraîne $p(v') \geq 0$. Par hypothèse de récurrence, v est dans $\mathcal{T}$, et $u = \text{Par}(v)$ aussi.

Sinon on note $v = u_0 \dots u_i$ et $w = u_{i+1} \dots u_n$. On a donc $u = vw$ et v, w de taille inférieure à n . De plus $p(v) = 0$ par construction et $p(w) = p(u) - p(v) = 0$ également. Considérons v' un préfixe de v , alors c'est un préfixe de u donc $p(v') \geq 0$. Considérons w' un préfixe de w , alors vw' est un préfixe de u donc $p(vw') \geq 0$, or $p(vw') = p(v) + p(w') = p(w')$.

On peut donc appliquer l'hypothèse de récurrence à w et v : v et w sont dans $\mathcal{T}$, et $u = \text{Concat}(v, w)$ aussi.

- Sinon u_0 est juste une lettre. Le mot $v = u_1 \dots u_n$ est de longueur inférieure à n et $p(v) = p(u) - p(u_0) = p(u) = 0$. De plus soit v' un préfixe de v , alors $u_0 v'$ est un préfixe de u , donc $p(u_0 v') = p(u_0 v') \geq 0$.

Par hypothèse de récurrence, $v \in \mathcal{T}$ et $u = \text{Concat}(u_0, v)$ aussi.

5. En déduire un algorithme permettant de tester si un mot est bien parenthésé.

On calcule $p(u)$ en sommant le p de tous les lettres, de gauche à droite. On vérifie que ça ne passe jamais dans le négatif, et que à la fin c'est nul.

Exercice 4 Des définitions de fonctions

Dans cet exercice on considère les entiers de Péano et les listes définies inductivement comme dans le cours.

Définir inductivement les fonctions suivantes :

1. la multiplication sur les entiers de Péano.

Ici on veut définir une application sur $\mathcal{T}^2$ et pas sur $\mathcal{T}$, mais le principe reste le même : il suffit de définir l'image selon la forme des arguments.

Pour définir la multiplication il nous faut d'abord la somme (comme dans le DM) :

$$\begin{aligned} \text{Add} \begin{cases} \text{Zero}, \text{Zero} & \mapsto \text{Zero} \\ \text{Zero}, \text{Successeur}(n) & \mapsto \text{Successeur}(n) \\ \text{Successeur}(n), \text{Zero} & \mapsto \text{Successeur}(n) \\ \text{Successeur}(n), \text{Successeur}(m) & \mapsto \text{Successeur}(\text{Add}(n, \text{Successeur}(m))) \end{cases} \\ \\ \text{Mult} \begin{cases} \text{Zero}, \text{Zero} & \mapsto \text{Zero} \\ \text{Zero}, \text{Successeur}(n) & \mapsto \text{Zero} \\ \text{Successeur}(n), \text{Zero} & \mapsto \text{Zero} \\ \text{Successeur}(n), \text{Successeur}(m) & \mapsto \text{Add}(m, \text{Mult}(n, \text{Successeur}(m))) \end{cases} \end{aligned}$$

2. la fonction factorielle sur les entiers de Péano.

$$\text{Fact} \begin{cases} \text{Zero} & \mapsto \text{Zero} \\ \text{Successeur}(n) & \mapsto \text{Mult}(\text{Successeur}(n), \text{Fact}(n)) \end{cases}$$

3. la fonction miroir qui renverse une liste.

Comme en ocaml ce n'est pas immédiat.

$$\begin{aligned} \text{Miroir} \begin{cases} [] & \mapsto [] \\ \text{Cons}_i(l) & \mapsto \text{Miroir} - \text{aux}_{\text{Cons}_i}([])(l) \end{cases} \\ \\ \text{Miroir} - \text{aux}_{l'} \begin{cases} [] & \mapsto l' \\ \text{Cons}_i(l) & \mapsto \text{Miroir} - \text{aux}_{\text{Cons}_i(l')}(l) \end{cases} \end{aligned}$$

4. la fonction max qui calcule le maximum d'une liste.

$$\text{Miroir} \begin{cases} [] & \mapsto 0 \text{ pas vraiment défini} \\ [i] & \mapsto i \\ \text{Cons}_i(l) & \mapsto \max(\text{Miroir}(l), i) \end{cases}$$

Exercice 5 Ordres bien fondés1. On va procéder par récurrence sur k (le nombre d'éléments dans chaque tuple).

Pour $k = 1$ $\mathbb{N}^1 = \mathbb{N}$, donc la propriété est vraie puisque l'ordre lexicographique sur $\mathbb{N}$ est l'ordre canonique sur $\mathbb{N}$, qui est bien fondé.

Supposons la propriété vraie pour un certain k . Montrons-la pour $k + 1$ par l'absurde : supposons l'existence d'une suite infinie strictement décroissante $(x_i)_{i \in I}$ de $\mathbb{N}^{k+1}$ pour l'ordre lexicographique.

On a deux cas possibles :

- Soit il existe une infinité de termes de la suite qui ont le même premier élément, c'est à dire qu'à partir d'un rang i_0 , tous les x_i ont le même premier élément.
On construit pour chaque x_i un x'_i qui est le tuple à k éléments formé de x_i privé de son premier élément. Alors la suite $(x'_i)_{i \in I, i \geq i_0}$ est une suite strictement décroissante de tuples à k éléments. Or on a par hypothèse de récurrence que l'ordre lexicographique sur $\mathbb{N}^k$ est bien fondé. Contradiction.
- Soit, en extrayant la suite $(y_i)_{i \in I}$ de tous les premiers éléments qui apparaissent dans la suite des x_i , on obtient une suite infinie strictement décroissante de $\mathbb{N}$. Contradiction.

On a donc montré par l'absurde que l'ordre lexicographique sur $\mathbb{N}^{k+1}$ est bien fondé.

Par principe de récurrence, les ordres lexicographiques sur $\mathbb{N}^k$ sont toujours bien fondés.

2. On montre les deux sens de l'équivalence :

Pour le sens direct, on suppose $\geq$ bien fondé et on procède par l'absurde. On suppose l'inverse de la propriété qu'on veut démontrer, c'est à dire qu'il existe $F \subset E$ tel que tout $y \in F$ n'est pas un minimum de F : il existe toujours un élément $x \in F$ tel que $x < y$.

On va construire une suite infinie strictement décroissante. On commence avec un x_0 quelconque dans F et on choisit successivement les x_i , pour $i \in \mathbb{N}$, de la manière suivante : si on a choisi x_i , alors d'après notre hypothèse il existe un élément $x \in F$ tel que $x < x_i$. On choisit alors $x_{i+1} = x$.

Par une récurrence immédiate par construction on peut montrer que cette construction nous fournit une suite infinie strictement décroissante de F , donc de E . Or on sait que $\geq$ est bien fondée sur E . Contradiction.

Pour le sens réciproque, supposons que toute partie F de E admette un élément minimal pour $\geq$. Supposons par l'absurde l'existence d'une suite infinie strictement décroissante pour $\geq$, notée $(x_i)_{i \in I}$. Alors l'ensemble $F = \{x_i | i \in I\}$ admet un minimum puisqu'il s'agit d'une partie de E . Ce minimum est un certain x_{i_0} et comme il est minimal, on peut dire que $x_{i_0+1} \geq x_{i_0}$, ce qui est absurde puisque notre suite est supposée strictement décroissante.

On a montré que $\geq$ est bien fondé.