

Induction



1 avril 2019

Damien Morard

Au cours de cette séance vous écrirez des définitions inductives et des preuves par récurrence, en plus de les implémenter.

Exercice 1 : Bien construire une preuve inductive (Rappel)

L'induction est un concept que nous découvrons souvent à partir des mathématiques. Il devient alors difficile de visualiser comment une preuve par induction fonctionne quand nous sortons des entiers. De plus la manière dont nous allons faire nos preuves par rapport à l'approche mathématique sera syntaxiquement différente.

Nous allons voir ensemble toutes les étapes nécessaires à la création d'une preuve, et comment exprimer les différents cas.

Point **important**, si vous avez bien assimilé la programmation logique, la déduction des cas en sera plus simple (ce sont nos **Faits**).

Détaillons ensemble les étapes nécessaires à une preuve par induction.

- Exprimer la propriété à prouver
- Déterminer et prouver le cas de base
- Déterminer le cas successeur
- Prouver le cas successeur à l'aide de l'hypothèse
- Supposer l'hypothèse vrai pour prouver le successeur
- Conclure

Il est essentiel de détailler et d'expliquer toutes vos étapes. Faites des phrases complètes ! Toutes vos preuves doivent avoir ce format !


Quand vous prouvez une propriété, il vous faut utiliser les **facts** et les **rules** sous la forme de règle d'inférence.

Exercice 2 : Implémentation de la définition inductive

Nous reprenons l'exercice sur les troupeaux sauf que vous implémenterez cette fois-ci. Le but est de vous donner l'idée de comment nous pouvons passer des règles d'inférences à la programmation logique.


Nous définissons ci-dessous un troupeau (qui est en réalité une file) de vaches.

Les vaches sont séparées par de l'herbe () . La file se termine par un arbre () .

—  \in troupeau

— $v \in \{ \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E}, \mathbf{F}, \dots, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{X}, \mathbf{Y}, \mathbf{Z} \} \implies v \text{  } t \in \text{troupeau}$

On considère la définition inductive suivante :

$$\frac{v \in \{ \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{E}, \mathbf{F}, \dots, \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{X}, \mathbf{Y}, \mathbf{Z} \} \quad t \in \text{troupeau}}{v \text{  } t \in \text{troupeau}}$$

$$\frac{\text{ \in \text{troupeau} \quad v \text{  } t \in \text{troupeau}}{\text{  t \in \text{troupeau}}$$

1. Implémenter à l'aide de LogicKit ou Prolog la définition d'un troupeau. Vous n'êtes pas obligé de mettre toutes les lettres pour les vaches.
2. Implémenter une règle pour compter le nombre de vache dans un troupeau.

Exercice 3 : Palindromes

Les palindromes¹ sont des mots qui peuvent se lire aussi bien de gauche à droite, que de droite à gauche. Par exemple (on ne tient pas compte ici des accents, des majuscules, des espaces ou de la ponctuation) :

- « Élu par cette crapule. »
- « Engage le jeu que je le gagne. »
- « À l'étape, épate-la ! »
- « Tu l'as trop écrasé, César, ce Port-Salut ! »

Nous souhaitons faire des palindromes de vaches **A** et **B** uniquement ; par exemple **AA**, **ABBA**, mais pas **ALETAPPEEPATELA**. (Palindromes composés uniquement de **A** et **B**. Il n'y a pas à tenir compte de la signification des mots.

1. Donnez une définition inductive de l'ensemble des palindromes **palyndromes** sur l'alphabet $\{ \mathbf{A}, \mathbf{B} \}$. Le mot vide s'écrit ϵ , et est évidemment un palindrome.
2. Donnez la définition de $nb_{\mathbf{A}} : \text{palindromes} \rightarrow \mathbb{N}$ qui retourne le nombre d'occurrences de la lettre **A** dans le palindrome. De même écrivez $nb_{\mathbf{B}}$ qui compte le nombre de **B**.
3. Prouvez qu'un palindrome ne peut contenir *à la fois* un nombre impair de **A** et un nombre impair de **B**. Pour cela, vous vous aiderez de $nb_{\mathbf{A}}$ et de $nb_{\mathbf{B}}$.
4. Implémenter la définition inductive du palindrome qui est donnée dans cet exercice.
5. Implémenter les règles nb_A et nb_B pour compter le nombre de *A* et de *B* dans notre palindrome.


1. <http://fr.wikipedia.org/wiki/Palindrome>

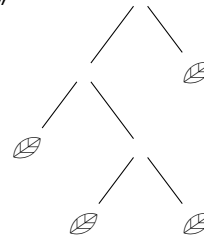
Exercice 4 : Arbres binaires

Les vaches aiment grimper aux arbres :

« Les pompiers de Saint-Benoît, en France, croyaient avoir tout vu et tout entendu lorsqu'ils ont reçu un appel pour le sauvetage d'une vache prise dans un arbre. La pauvre bête qui s'était aventurée le long d'une paroi rocheuse s'est ainsi retrouvée dans un arbre après avoir glissé.

Une équipe spéciale a été dépêchée sur place pour tenter de sauver la vache. Mais cette dernière, paniquée et déséquilibrée, a lourdement chuté de son perchoir. Elle n'a pu être sauvée à temps. »

Les arbres binaires sont des structures très utilisées en informatique, car elles permettent de rechercher rapidement des données. Un arbre est une feuille, notée , ou bien un nœud $\langle g, d \rangle$ dont les deux successeurs g et d sont des arbres. Un exemple d'arbre binaire vous est donné ci-dessous.



1. Donnez la définition inductive de A , l'ensemble des arbres binaires.
2. Prouvez que le nombre $n(a)$ de nœuds d'un arbre a est donné par la formule $1 + n(g) + n(d)$, où g et d sont les deux fils de l'arbre a .
3. Prouvez que la hauteur $h(a)$ d'un arbre a est donnée par la formule $1 + \max(h(g), h(d))$, où g et d sont les deux fils de l'arbre a .
4. Prouvez que la hauteur d'un arbre est toujours inférieure ou égale à son nombre de nœuds.
5. Implémenter la définition inductive de l'arbre binaire de l'exercice.
6. Implémenter la règle pour calculer le nombre de noeuds dans l'arbre a , $n(a)$.
7. Implémenter la règle pour calculer la hauteur d'un arbre a , $h(a)$.

