Définitions Inductives et preuves par induction

Définitions inductives en informatique

- Syntaxe concrete
- Syntaxe abstraite
- Règles de typage
- Règles d'évaluation

Le principe

Une définition inductive est caracterisée par :

- Une ou plusieures assertions
- Un ensemble de règles d'inférence pour dériver ces assertions

Exemple :

- Assertion : "`X est naturel`" ou "`X nat`"
- Règles d'inférence :

  R1 : `0 est naturel`

  R2 : Si `n est naturel`, alors `succ(n) est naturel`.

Notation

Les règles d'inférence sont notées

$$\frac{\texttt{Hypothèse}_1 \ldots \texttt{Hypothèse}_n}{\texttt{Conclusion}} \text{(Nom de la règle)}$$

- `Conclusion` est une assertion
- $\texttt{Hypothèse}_1 \ldots \texttt{Hypothèse}_n$ sont des assertions
- En général $n \geq 0$. Si $n = 0$ la règle est un axiome

Exemple (règle unaire)

Les entiers naturels

$$\frac{}{0 \texttt{ est naturel}}\,(Nat0) \qquad \frac{n \texttt{ est naturel}}{succ(n) \texttt{ est naturel}}\,(Nat+)$$

Exemple (règle binaire)

Les arbres binaires

$$\frac{}{vide \texttt{ est un arbre binaire}}\,\text{(Abin-nil)}$$

$$\frac{A_1 \texttt{ est un arbre binaire} \qquad A_2 \texttt{ est un arbre binaire}}{node(A_1, A_2) \texttt{ est un arbre binaire}}\,\text{(Abin-ind)}$$

Exemple

Les mots sur un alphabet A

$$\frac{}{\epsilon \texttt{ mot}} \qquad \frac{a \in A \quad n \texttt{ mot}}{a.n \texttt{ mot}}$$

Exemple (plusieurs axiomes, règles unaires et binaires)

Les expressions de la logique propositionnelle sur l'alphabet A

$$\frac{p \in A}{p \texttt{ expr}}$$

$$\frac{A_1 \texttt{ expr} \qquad A_2 \texttt{ expr}}{A_1 \vee A_2 \texttt{ expr}} \qquad \frac{A_1 \texttt{ expr} \qquad A_2 \texttt{ expr}}{A_1 \wedge A_2 \texttt{ expr}}$$

$$\frac{A_1 \texttt{ expr} \qquad A_2 \texttt{ expr}}{A_1 \rightarrow A_2 \texttt{ expr}} \qquad \frac{A \texttt{ expr}}{\neg A \texttt{ expr}}$$

Exemple (plusieures assertions)

Les forêts

$$\frac{}{vide\ \texttt{arbre}} \qquad \frac{}{nil\ \texttt{foret}}$$

$$\frac{A\ \texttt{arbre} \qquad f\ \texttt{foret}}{cons(A, f)\ \texttt{foret}} \qquad \frac{f\ \texttt{foret}}{node(f)\ \texttt{arbre}}$$

Dérivation d'une assertion

Une assertion A est dérivable ssi

– A est un axiome

$$\frac{}{A}$$

– ou il y a une règle de la forme

$$\frac{A_1 \qquad A_n}{A}$$

telle que $A_1, \ldots, A_n$ sont dérivables

Ensemble inductif

Un ensemble inductif est le plus petit ensemble engendré par un système de règles d'inférence.

Preuves par Induction

Preuves par induction

- Induction sur les entiers
  - Induction mathématique
  - Induction complète
  - Équivalence
- Induction bien fondée
- Induction structurelle
- Induction sur un ensemble inductif

Induction sur les entiers I (induction mathématique)

Théorème : Soit P une propriété sur les entiers. Supposons

(IM1) $P(0)$,

(IM2) $\forall n \in \mathbb{N}. P(n) \rightarrow P(n+1)$,

alors $\forall n \in \mathbb{N}. P(n)$

Exemples

$$1) \quad \sum_{i=1}^{n} i = \frac{n * (n+1)}{2} \qquad 2) \quad n^2 = \sum_{i=1}^{n} (2i-1)$$

Mais comment prouver

1. "Tout entier est décomposable en produit de nombres premiers"
2. "Si n est divisible par 3, alors $fib(n)$ est pair, sinon $fib(n)$ est impair".

Induction sur les entiers II (induction complète)

Théorème : Soit P une propriété sur les entiers. Supposons

$$(IC) \qquad \forall n \in \mathbb{N}. ((\forall k < n. P(k)) \rightarrow P(n))$$

alors $\forall n \in \mathbb{N}. P(n)$

Équivalence des deux principes

Malgré l'apparente supériorité du deuxième principe, on prouve

Théorème : Induction mathématique et complète sont équivalentes.

Théorème fondamental du cours

Théorème : Tous le monde est d'accord avec le professeur.

Preuve : On montre, par induction sur le nombre de personnes dans l'amphi, que tout groupe de n personnes contenant le professeur est d'accord avec lui.
Cas de base : il y a seulement le professeur, trivial.
Cas inductif : on suppose l'enoncé vrai pour tout groupe de n personnes, et on le prouve pour tout groupe de $n+1$.
Numérotons de 1 à $n+1$ les personnes en question, de façon que le professeur soit le numéro n, et considérons le groupe A des premières n et le groupe B des dernières n personnes.
Les deux groupes contiennent le professeur et sont de taille

$n < n+1$, donc on peut appliquer l'hypothèse d'induction et en déduire qu'ils sont tous d'accord avec le professeur (qui est dans les deux), ce qui nous permet de conclure.

Corollaire : Le professeur a toujours raison.

Principe d'induction bien fondée

Un ensemble $\mathcal{A}$, un ordre strict $>$ et une propriété P sur $\mathcal{A}$

Principe d'induction :

Si

1. "pour tout élément minimal $y \in \mathcal{A}$ on a $P(y)$"
2. "le fait que $P(z)$ soit vérifiée pour tout élément $z<x$ implique $P(x)$"

alors

"pour tout $x \in \mathcal{A}$ on a $P(x)$"

Ce principe est-il toujours bien défini ?

Théorème :

Si $>$ est bien fondé, alors le principe d'induction est correct.

Théorème :

Si le principe d'induction est correct, alors $>$ est bien fondé.

Corollaire : Le principe d'induction est correct pour les ensembles inductifs.

Corollaire : Le principe d'induction structurelle.

Exemples

- Les mots :
  $P(m)$ est la propriété :
  $concat(concat(m, v_1), v_2), = concat(m, concat(v_1, v_2))$
- Les arbres binaires :
  $P(a)$ est la propriété : $feuilles(a) = noeuds\_internes(a) + 1$

Induction sur d'autres ordres bien fondés

- Ordre lexicographique
- Ordre multi-ensemble
- Combinaisons

Ordres lexicographiques

Soit $>_{A_i}$ un ordre strict sur l'ensemble $\mathcal{A}_i$.

Ordre lexicographique sur le produit de 2 ensembles :

$$(x, y) >_{lex} (x', y') \text{ ssi } (x >_{A_1} x') \text{ ou } (x = x' \text{ et } y >_{A_2} y')$$

Exemple :

$(4, "abc") >_{lex} (3, "abc") >_{lex} (2, "abcde") >_{lex} (2, "bcde") >_{lex} (2, "e") >_{lex} (1, "e") >_{lex} (0, \epsilon)$

Ordre lexicographique sur le produit de n ensembles

Si chaque $>_{A_i}$ est un ordre strict sur l'ensemble $\mathcal{A}_i$, alors $>_{lex}$ est un ordre strict qui permet de comparer deux n-uplets de la manière suivante :

$$(x_1, \ldots, x_n) >_{lex} (x'_1, \ldots, x'_n) \text{ ssi } \quad \exists 1 \leq j \leq n \quad (x_j >_{A_j} x'_j \text{ and } \forall 1 \leq i < j \;\; x_i = x'_i)$$

Théorème : Si chaque $>_{A_i}$ est un ordre strict sur $\mathcal{A}_i$, alors l'ordre lexicographique $>_{lex}$ sur le produit de $\mathcal{A}_1 \times \ldots \times \mathcal{A}_n$ est un ordre strict sur $\mathcal{A}_1 \times \ldots \times \mathcal{A}_n$.

Avertissement : $>_{lex}$ n'est pas l'ordre du dictionaire ! !

Exemple : la fonction d'Ackerman

Montrer par induction que la fonction suivante termine.

Ackerman(0,n)	=	n+1
Ackerman(m+1,0)	=	Ackerman(m,1)
Ackerman(m+1,n+1)	=	Ackerman(m,Ackerman(m+1,n))

Les multi-ensembles

Définition : Soit $\mathcal{A}$ un ensemble. Un multi-ensemble de base $\mathcal{A}$ est une fonction $\mathcal{M} : \mathcal{A} \to \mathbb{N}$. Le multi-ensemble $\mathcal{M}$ est fini si $\mathcal{M}(x) > 0$ seulement pour un nombre fini d'éléments de $\mathcal{A}$.

Notation : $\{\!\{a, a, b\}\!\}$.

Ordres multi-ensembles

Définition : $\mathcal{M} >_{mul} \mathcal{N}$ ssi $\mathcal{N}$ s'obtient à partir de $\mathcal{M}$ en appliquant la règle suivante un nombre fini de fois : enlever un élément x de $\mathcal{M}$ et le remplacer par un nombre fini d'éléments plus petits que x (par rapport à l'ordre $>$).

Notation :

$$\{\!\{5, 3, 1, 1\}\!\}$$

Exemple :

$$\{\!\{5, 3, 1, 1\}\!\} >_{mul} \{\!\{4, 3, 3, 1\}\!\}$$

Théorème : Si $>_A$ est un ordre strict sur $\mathcal{A}$, alors $>_{mul}$ est un ordre strict sur les multi-ensembles de base $\mathcal{A}$.

Exemple

Un homme possède une somme d'argent en euros. Chaque jour il procède de la façon suivante :

– soit il jette une pièce de monnaie dans une fontaine,
– ou bien il change l'un de ses billets à la banque.

Montrer que ce processus termine, c'est à dire, que dans un temps fini l'homme reste sans argent.