Logique

La logique

Étude du raisonnement (des démonstrations)

Démonstration : objet qui permet de juger de la vérité d'une proposition

Tour à tour branche de la philosophie, des mathématiques, puis de l'informatique

Objectifs

Un cours introductif principalement destiné aux informaticiens (mais qui peut aussi être utilisé par des philosophes et des mathématiciens)

Présenter

- les notions de langage, de démonstration, d'algorithme, de modèle et d'ensemble
- des exemples de théories : arithmétique, théorie des ensembles...
- les théorèmes de complétude (Gödel), d'indécidabilité (Church), d'incomplétude (Gödel) et d'élimination des coupures (Gentzen)
- des méthodes de démonstration automatique

Quel rapport entre la logique (les démonstrations, la vérité...) et l'informatique ?

1. Les ordinateurs sont des machines à vérité

Un programme qui calcule la centième décimale de π
Son exécution nous apprend que c'est un 9
Jugement de la vérité de la proposition « la centième décimale de π est un 9 »

Même nature qu'une démonstration ?
Le programme produit un résultat : 9, mais peut-il aussi produire une démonstration de la proposition « la centième décimale de π est un 9 » ?

2. La vérification et la recherche de démonstrations

La logique : base des logiciels

- qui vérifient la correction des démonstrations
- qui recherchent des démonstrations

3. Démontrer la correction d'algorithmes et de programmes

Systèmes critiques : transports, énergie, médecine...

Éviter les bugs

Démontrer que ces programmes sont corrects

Programmes : faire ceci, faire cela... mais dans quel but ?

Plus conceptuellement

La logique : langage, démonstration, algorithme, modèle, ensemble

L'informatique : langage, algorithme, information, machine

La notion de définition inductive

La notion de langage

I. La notion de définition inductive

Définir l'ensemble des propositions

Définir le sous-ensemble des propositions démontrables

Comment définir un ensemble ou une relation ?

Par une définition explicite :
$\{x \in \mathbb{N} \mid \exists z \in \mathbb{N}\ x = 2 \times z\}$
$\{\langle x, y\rangle \in \mathbb{N}^2 \mid \exists z \in \mathbb{N}\ x = y \times z\}$

Mais ça ne suffit pas...

Par une définition inductive
La notion de définition inductive : le théorème du point fixe

E, $\leq$ relation d'ordre

Par exemple, $\mathbb{R}, \leq$

Par exemple, $[0,1], \leq$

Par exemple, $\mathcal{P}(A)$, $\subseteq$ (A ensemble quelconque)

Le premier théorème du point fixe

u_0, u_1, ... suite croissante
l limite de $(u_i)_i$ si $l = sup\ \{u_0, u_1, ...\}$

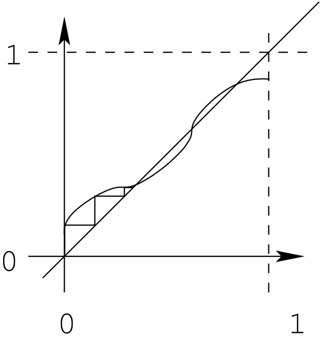
$E, \leq$ faiblement complète si toute suite croissante a une limite

$[0, 1], \leq$ faiblement complète, $\mathcal{P}(A), \subseteq$ faiblement complète, mais pas $\mathbb{R}^+, \leq$

f croissante est continue si $lim_i\ (f\ \ u_i) = f\ \ (lim_i\ \ u_i)$

Théorème : $\leq$ faiblement complète, un minimum m et f continue alors f a un point fixe

Le plus petit point fixe est $lim_i\ (f^i\ \ m)$


1
0
0
1

Le second théorème du point fixe

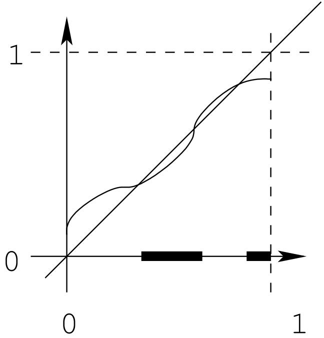
Pour les fonctions croissantes (mais non nécessairement continues)

$E, \leq$ fortement complète si tout sous-ensemble de E a une borne supérieure
Donc tout sous-ensemble de E a une borne inférieure

$[0,1], \leq$ fortement complète, $\mathcal{P}(A), \subseteq$ fortement complète, mais pas $\mathbb{R}^+, \leq$, ni $\mathbb{R}^-, \leq$

Théorème : $\leq$ fortement complète et f croissante alors f a un point fixe

Le plus petit point fixe est $\mathit{inf}\ \{c \mid fc \leq c\}$



1
0
0
1

Une première définition inductive

$P = 2\mathbb{N}$ est défini par
$0 \in P$ et si $n \in P$ alors $n + 2 \in P$

$$\overline{0}$$

$$\frac{n}{n+2}$$

$$\overline{0 \in P}$$

$$\frac{n \in P}{n + 2 \in P}$$

P n'est pas le seul ensemble qui contient 0 et qui est clos par la fonction $n \mapsto n + 2$

Mais c'est le plus petit de ces ensembles

F de $\mathcal{P}(\mathbb{N})$ dans $\mathcal{P}(\mathbb{N})$
$F(A) = \{0\} \cup \{x + 2 \mid x \in A\}$
F croissante et continue
(A contient 0 et clos par $n \mapsto n + 2$) : $F(A) \subseteq A$
P est défini comme le plus petit point fixe de F

Second théorème du point fixe : c'est l'intersection de tous les ensembles qui contiennent 0 et qui sont clos par $n \mapsto n + 2$
Premier théorème du point fixe : c'est la réunion de $\varnothing$, $F(\varnothing)$, $F(F(\varnothing))$...

Cas général

Un ensemble E
On définit un sous-ensemble B de E
par des fonctions de fermeture (règles) f_1, f_2...

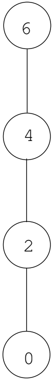
$F(A) = \bigcup_i \{f_i(a_1, ..., a_{n_i}) \mid a_1, ..., a_{n_i} \in A\}$
F croissante et continue
B est le plus petit point fixe de F

La notion de dérivation

$x \in B$ si $x \in F^k(\varnothing)$ pour un certain k

c'est-à-dire, s'il existe i, $y_1, ..., y_n \in F^{k-1}(\varnothing)$ tels que $x = f_i(y_1, ..., y_n)$

Par récurrence sur k si $x \in B$ alors il existe un arbre dont les nœuds sont étiquetés par des éléments de E et les enfants d'un nœud x sont $y_1, ..., y_n$ tq il existe une règle f_i tq $x = f_i(y_1, ..., y_n)$



$$\dfrac{}{0} \qquad \dfrac{}{0 \in P}$$
$$\dfrac{}{2} \qquad \dfrac{}{2 \in P}$$
$$\dfrac{}{4} \qquad \dfrac{}{4 \in P}$$
$$\dfrac{}{6} \qquad \dfrac{}{6 \in P}$$

Dérivations étiquetées par les éléments et par les noms

$\overline{0}$	$-\ z$	$\overline{0}\ z$
$\overline{2}$	$-\ p$	$\overline{2}\ p$
$\overline{4}$	$-\ p$	$\overline{4}\ p$
$\overline{6}$	$-\ p$	$\overline{6}\ p$

Exemple

$E = \{a, b\}^*$

$$\overline{b}$$

$$\frac{X}{a \; X \; a}$$

Exemple

$$\frac{A \wedge B}{A}$$

$$\frac{A \wedge B}{B}$$

$$\frac{A \Rightarrow B \quad A}{B}$$

$$\frac{}{(P \Rightarrow Q) \wedge P}$$

$$\dfrac{\dfrac{\overline{(P \Rightarrow Q) \land P}}{P \Rightarrow Q} \qquad\qquad \dfrac{\overline{(P \Rightarrow Q) \land P}}{P}}{Q}$$

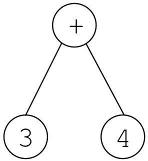
II. La notion de langage

Une notion très générale

Langage logique, langage de programmation, langage de requête...

On oublie la contrainte de linéarité
On ne s'intéresse pas à savoir si on écrit $3+4$, $+(3,4)$ ou $34+$

Les expressions sont des arbres

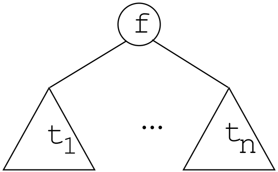


Les langages sans variables

Un langage (sans variables) est un ensemble de symboles, chacun muni d'un nombre entier appelé son arité ou nombre d'arguments

L'ensemble des expressions du langage est l'ensemble d'arbres défini inductivement par la règle

$$\frac{t_1 \qquad \ldots \qquad t_n}{f(t_1, \ldots, t_n)} \text{ si } f \text{ est un symbole d'arité } n$$



Exemple

Une constante (c'est-à-dire un symbole d'arité nulle) 0
Un symbole unaire S
Deux symboles binaires $+$, $\times$
Deux symboles unaires $pair$, $impair$
Un symbole binaire $\Rightarrow$

$$impair(S(S(S(0)))) \Rightarrow pair(S(S(S(S(0)))))$$

Si un nombre est impair alors son successeur est pair

$$\forall x \ (impair(x) \Rightarrow pair(S(x)))$$

Des variables
Des symboles qui lient des variables

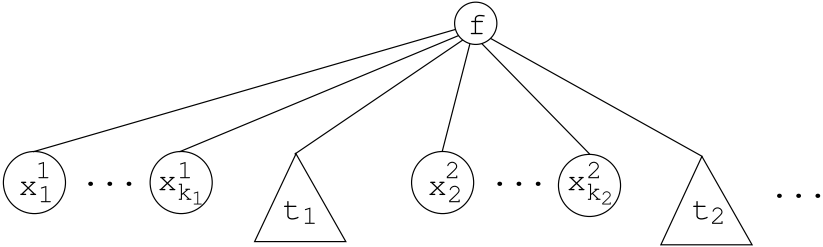
Les langages avec variables

L'arité d'un symbole est un n-uplet $\langle k_1, ..., k_n \rangle$
le symbole a n arguments, il lie k_1 variables dans le premier, ..., k_n variables dans le $n^{\text{ème}}$
Exemple : $\forall$ a l'arité $\langle 1 \rangle$
Un ensemble de symboles et un ensemble infini de variables

Les expressions sont définies inductivement par les règles :

- les variables sont des expressions
- si f est un symbole d'arité $\langle 1, 3 \rangle$, t et u sont des expressions, w, x, y, z sont des variables alors $f(w\ t, x\ y\ z\ u)$ est une expression (à généraliser)

$f(x_1^1 \ldots x_{k_1}^1\, t_1, \ldots, x_1^n \ldots x_{k_n}^n\, t_n)$ est l'arbre



Les variables et les variables libres

- $Var(x) = \{x\}$
- $Var(f(x_1^1 \ldots x_{k_1}^1 t_1, \ldots, x_1^n \ldots x_{k_n}^n t_n))$
  $= Var(t_1) \cup \{x_1^1, \ldots, x_{k_1}^1\} \cup \cdots \cup Var(t_n) \cup \{x_n^n, \ldots, x_{k_n}^n\}$

- $VL(x) = \{x\}$
- $VL(f(x_1^1 \ldots x_{k_1}^1 t_1, \ldots, x_1^n \ldots x_{k_n}^n t_n))$
  $= (VL(t_1) \setminus \{x_1^1, \ldots, x_{k_1}^1\}) \cup \cdots \cup (VL(t_n) \setminus \{x_n^n, \ldots, x_{k_n}^n\})$

Exemple

$$Var(\forall x\ (x = x)) = \{x\}$$

$$VL(\forall x\ (x = x)) = \varnothing$$

$$VL(\forall x\ (x = y)) = \{y\}$$

Les langages à plusieurs sortes d'objets

0, S, $+$, $\times$, $pair$, $impair$, $\Rightarrow$, $\forall$

Empêcher $0 \Rightarrow 0$

Distinguer 0, $S(0)$, $S(x)$... termes
de $pair(0)$, $impair(0)$, $\forall x\ (pair(x))$... propositions

Mais aussi peut-être les termes de vecteurs, les termes de scalaires...

Les langages à plusieurs sortes d'objets

Un ensemble de sortes $\{Terme, Prop\}$ plus généralement $\mathcal{S}$

L'arité d'un symbole est un $n+1$-uplet de sortes $\langle s_1, ..., s_n, s' \rangle$

Si t_1 terme de sorte s_1, t_2 terme de sorte s_2, ..., t_n terme de sorte s_n et f d'arité $\langle s_1, ..., s_n, s' \rangle$ alors $f(t_1, ..., t_n)$ de sorte s'

Plusieurs sortes d'objets + variables

$$\langle\langle s_1^1, ..., s_{k_1}^1, s'^1\rangle, ..., \langle s_1^n, ..., s_{k_n}^n, s'^n\rangle, s''\rangle$$

Exemple $\forall$ d'arité $\langle\langle \mathit{Terme}, \mathit{Prop}\rangle, \mathit{Prop}\rangle$

La prochaine fois

La logique des prédicats