second order propositional logic

logical verification

week 11

2004 11 24

the course

1st order **propositional** logic	$\leftrightarrow$	simple type theory
		$\lambda\!\rightarrow$
1st order **predicate** logic	$\leftrightarrow$	type theory with **dependent types**
		λP
2nd order propositional logic	$\leftrightarrow$	**polymorphic** type theory
		$\lambda 2$

propositional logic

$a \ b \ c \ \ldots$

$A \longrightarrow B$

$\bot$

$\top$

$\neg A$

$A \wedge B$

$A \vee B$

predicate logic

$x \; y \; z \; \ldots$

$a(\ldots) \; b(\ldots) \; c(\ldots) \; \ldots \qquad f(\ldots) \; g(\ldots) \; h(\ldots) \; \ldots$

$A \rightarrow B$

$\bot$

$\top$

$\neg A$

$A \wedge B$

$A \vee B$

$\forall x \, . \, A$

$\exists \, x \, . \, A$

second order propositional logic

$a \; b \; c \; \ldots$

$A \to B$

$\bot$

$\top$

$\neg A$

$A \wedge B$

$A \vee B$

$\forall a \,.\, A$

$\exists \, a \,.\, A$

example

$$a \longrightarrow a$$

$$\forall a.\, a \longrightarrow a$$

if it's tuesday, then it's tuesday

for every proposition, that proposition implies itself

the rules

introduction rules	elimination rules
$I[x]{\rightarrow}$	$E{\rightarrow}$
	$E\bot$
$I\top$	
$I[x]\neg$	$E\neg$
$I\wedge$	$El\wedge \quad Er\wedge$
$Il\vee \quad Ir\vee$	$E\vee$
$I\forall$	$E\forall$
$I\exists$	$E\exists$

propositional logic: rules for implication

implication introduction

$$\dfrac{\begin{array}{c}[A^x]\\ \vdots\\ B\end{array}}{A \to B}\ I[x]{\to}$$

implication elimination

$$\dfrac{\begin{array}{c}\vdots\\ A \to B\end{array}\qquad\qquad \begin{array}{c}\vdots\\ A\end{array}}{B}\ E{\to}$$

propositional logic: rules for falsum and truth

falsum elimination

$$\dfrac{\begin{array}{c}\vdots \\ \bot\end{array}}{A}\ E\bot$$

truth introduction

$$\dfrac{}{\top}\ I\top$$

propositional logic: rules for conjunction

conjunction introduction

$$\dfrac{\begin{matrix}\vdots \\ A\end{matrix} \qquad \begin{matrix}\vdots \\ B\end{matrix}}{A \land B}\ I\land$$

conjunction elimination

$$\dfrac{\begin{matrix}\vdots \\ A \land B\end{matrix}}{A}\ El\land \qquad \dfrac{\begin{matrix}\vdots \\ A \land B\end{matrix}}{B}\ Er\land$$

propositional logic: rules for disjunction

disjunction introduction

$$\dfrac{\begin{matrix}\vdots \\ A\end{matrix}}{A \vee B}\ Il\vee \qquad \dfrac{\begin{matrix}\vdots \\ B\end{matrix}}{A \vee B}\ Il\vee$$

disjunction elimination

$$\dfrac{\begin{matrix}\vdots \\ A \vee B\end{matrix} \qquad \begin{matrix}\vdots \\ A \rightarrow C\end{matrix} \qquad \begin{matrix}\vdots \\ B \rightarrow C\end{matrix}}{C}\ E\vee$$

2nd order propositional logic: rules for universal quantification

universal quantification introduction

$$\frac{\begin{array}{c}\vdots \\ A\end{array}}{\forall a.\, A}\ I\forall$$

variable condition: a not a free variable in any open assumption

universal quantification elimination

$$\frac{\begin{array}{c}\vdots \\ \forall a.\, A\end{array}}{A[a := B]}\ E\forall$$

2nd order propositional logic: rules for existential quantification

existential quantifier introduction

$$\dfrac{\begin{matrix}\vdots \\ A[a := B]\end{matrix}}{\exists a.\, A}\; I\exists$$

existential quantifier elimination

$$\dfrac{\begin{matrix}\vdots \\ \exists a.\, A\end{matrix} \qquad \begin{matrix}\vdots \\ \forall a.\, (A \to B)\end{matrix}}{B}\; E\exists$$

variable condition: a not a free variable in B

variable conditions

- for rule $I\forall$

  check:

  variable does not occur in any of the available assumptions

- for rule $E\exists$

  check:

  variable does not occur in the conclusion

example 1

$$(\forall b.\, b) \to a$$

example 2

$$a \to \forall b. \, ((a \to b) \to b)$$

example 3

$$(\exists b.\, a) \rightarrow a$$

example 4

$$\exists b.((a \longrightarrow b) \vee (b \longrightarrow a))$$

example 5

$$\forall a.\, \forall b.\, ((a \rightarrow b) \vee (b \rightarrow a))$$

this needs classical logic

$$\forall a.\, (a \vee \neg a)$$

non-example 6

$$a \longrightarrow \forall a.\, a$$

non-example 7

$$(\exists a.\, a) \to a$$

higher order logic

the ‘order’ of a variable

first order	object
second order	set of objects
	predicate on objects
	function from objects to objects
third order	set of second order objects
	predicate on predicates on objects
	function from second order objects to . . .

etc.

example from 2nd order predicate logic

induction principle for natural numbers

$$\forall a.\ \big(a(0) \to (\forall m.\ a(m) \to a(S(m))) \to \forall n.\ a(n)\big)$$

m	1st order variable
n	1st order variable
0	1st order constant
a	2nd order variable
S	2nd order constant

only predicates without arguments

quantify over predicates	$\longrightarrow$	**2nd order** predicate logic
. . . the same without terms	$\longrightarrow$	2nd order **propositional** logic

the connectives in Coq

$\to$	hard-wired into the type theory
$\forall$	hard-wired into the type theory
$\bot$	inductive type
$\wedge$	inductive type
$\vee$	inductive type
$\exists$	inductive type

inductive definition of False

```
Inductive False : Prop :=
  .
```

$$\texttt{False\_ind} \;:\; \forall a.\,\bot \to a$$

the constructors are the introduction rules

the induction principle is the elimination rule

inductive definition of and

```
Inductive and (a b : Prop) : Prop :=
  conj : a → b → a ∧ b .
```

$$\texttt{and\_ind} \ : \ \forall a\, b\, c.\, (a \to b \to c) \to (a \wedge b) \to c$$

the constructor is the introduction rule

the induction principle gives the elimination rules

alternative version of conjunction elimination

conjunction elimination: alternative version

$$\dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix} \qquad \begin{matrix}\vdots \\ A \to B \to C\end{matrix}}{C}\ E\wedge$$

conjunction elimination: normal version

$$\dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix}}{A}\ El\wedge \qquad \dfrac{\begin{matrix}\vdots \\ A \wedge B\end{matrix}}{B}\ Er\wedge$$

inductive definition of `or`

```
Inductive or (a b : Prop) : Prop :=
  or_introl : a → a ∨ b
| or_intror : b → a ∨ b .
```

$$\texttt{or\_ind} \;:\; \forall a\, b\, c.\, (a \to c) \to (b \to c) \to (a \vee b) \to c$$

the constructors are the introduction rules

the induction principle is the elimination rule

impredicative definition of `False`

$$\bot \quad := \quad \forall a.\, a$$

induction principle next to impredicative definition

$$\forall a.\, \bot \to a$$
$$\forall a.\, \qquad a$$

impredicative definition of and

$$a \wedge b \quad := \quad \forall c.\, (a \to b \to c) \to c$$

induction principle next to impredicative definition

$$\forall a\ b.\ \forall c.\, (a \wedge b) \to (a \to b \to c) \to c$$

$$\forall c. \qquad\qquad\quad (a \to b \to c) \to c$$

impredicative definition of `or`

$$a \vee b \quad := \quad \forall c.\, (a \to c) \to (b \to c) \to c$$

induction principle next to impredicative definition

$$\begin{aligned} \forall a\ b.\ \forall c.\ (a \vee b) &\to (a \to c) \to (b \to c) \to c \\ \forall c.\ \phantom{(a \vee b) \to{}} & \phantom{\to{}} (a \to c) \to (b \to c) \to c \end{aligned}$$

impredicative definition of the booleans

$$\forall a.\, a \longrightarrow a \longrightarrow a$$

impredicative definition of the natural numbers

$$\forall a.\, a \to (a \to a) \to a$$

why have inductive types as primitive then?

- one can prove less equalities
- one gets weaker induction principles
- **some** people don't like impredicativity