Mathematics for Computer Scientists 2 (G52MC2)

L06 : More predicate logic

Thorsten Altenkirch

School of Computer Science
University of Nottingham

October 22, 2009

Frege

Friedrich Ludwig Gottlob Frege (1848-1925)

- Frege developed predicate logic in the 1870ies.
- He also introduced a graphic notation system nobody is using anymore ($\vdash$ for provable is a relict).
- Frege wrote a book called *Begriffsschrift* with the goal to make the logical foundations of Mathematics precise.
- Unluckily, there was a serious bug, which was discovered by Betrand Russell. Frege's theory was inconsistent.
- Later predicate logic was used for:

  Arithmetic The theory of the natural numbers.

  ZF Set Theory The classical theory of sets.

Predicate logic (in Coq)

Sets or *types*.
(Not exactly the same sets as in ZF set theory).

Terms or *expressions*. Made from constants, variables and functions.

Propositions Propositional logic + quantifiers ($\forall$,$\exists$).

Predicates and relations Functions into **Prop**. If they have more than ane argument they are called relations. Equality ($=$) is a special relation.

Summary of tactics

connective	Introduction	Elimination
$P \to Q$	`intro(s)`	`apply` *Hyp*
$P \wedge Q$	`split`	`elim` *Hyp*
True	`split`	
$P \vee Q$	`left,right`	`case` *Hyp*
False		`case` *Hyp*
forall $x : A, P$	`intro(s)`	`apply` *Hyp*
exists $x : A, P$	`exists` *wit*	`elim` *Hyp*
$a = b$	`reflexivity`	`rewrite` *Hyp*

Examples of tautologies

- $(\forall x : D.P\,x \wedge Q\,x) \leftrightarrow (\forall y : D, P\,y) \wedge (\forall y : D, Q\,y)$
- $(\exists x : D.P\,x \vee Q\,x) \leftrightarrow (\exists y : D, P\,y) \vee (\exists y : D, Q\,y)$
- $(\forall x : D.P\,x \rightarrow R) \leftrightarrow (\exists x : D.P\,x) \rightarrow R$
- $(\forall x : D.\neg P\,x) \leftrightarrow \neg\exists x : D, P\,x$
- $(\exists x : D.\neg P\,x) \rightarrow \neg\forall x : D, P\,x$

Examples of classical tautologies

The following require the principle of the excluded middle (`Import Classical`):

- $(\neg \forall x : D, P\, x) \to \exists x : D. \neg P\, x$
- $\forall x : D, P\, x \lor \neg P\, x$

Decidability

We say a logical system is **decidable**, if there is a computer program which can determine wether a proposition is true (or provable) or not.

Decidable:

- Propositional logic
- Predicate logic over Bool

Undecidable:

- General predicate logic (with predicate and set variables)

However, there are some heuristics which work in many cases (In Coq: `firstorder`). **Not for the homework!**