Semantics of Programming Languages (10)

Christian Urban

http://www4.in.tum.de/~urbanc/Teaching/semantics08.html
lemma shows "(rev pi) \circ (pi \circ x) = x"

proof (induct pi)
  case Nil
  show "(rev []) \circ ([] \circ x) = x" by simp
next
  case (Cons a pi)
  have ih: "(rev pi) \circ (pi \circ x) = x" by fact
  obtain z y where "a = (z,y)" by (cases a) (simp)
  ...

\begin{itemize}
  \item thm prod.exhaust
\end{itemize}

(\forall a b. y = (a, b) \implies P) \implies P
Substitution Lemma: If \( x \not\equiv y \) and \( x \not\in \text{fv}(L) \), then
\[
M[x := N][y := L] \equiv M[y := L][x := N[y := L]]
\]

Proof: By induction on the structure of \( M \).

- **Case 1:** \( M \) is a variable.
  - Case 1.1. \( M \equiv x \). Then both sides equal \( N[y := L] \) since \( x \not\equiv y \).
  - Case 1.2. \( M \equiv y \). Then both sides equal \( L \), for \( x \not\in \text{fv}(L) \) implies \( L[x := \ldots] \equiv L \).
  - Case 1.3. \( M \equiv z \not\equiv x, y \). Then both sides equal \( z \).

- **Case 2:** \( M \equiv \lambda z. M_1 \). By the variable convention we may assume that \( z \not\equiv x, y \) and \( z \) is not free in \( N, L \).
  \[
  (\lambda z. M_1)[x := N][y := L] \equiv \lambda z.(M_1[x := N][y := L]) \\
  \equiv \lambda z.(M_1[y := L][x := N[y := L]]) \\
  \equiv (\lambda z. M_1)[y := L][x := N[y := L]].
  \]

- **Case 3:** \( M \equiv M_1 M_2 \). The statement follows again from the induction hypothesis.
“The method of ‘postulating’ what we want has many advantages; they are the same as the advantages of theft over honest toil.”

B. Russell, Introduction of Mathematical Philosophy
Induction Principles

\[
\begin{align*}
\forall \text{name}. \ P (\text{Var name}) \\
\forall \text{t1 t2. } [P \text{ t1}; P \text{ t2}] &\implies P (\text{App t1 t2}) \\
\forall \text{name t. } P \text{ t} &\implies P \text{ Lam [name].t} \\
\hline \\
\hline \\
\end{align*}
\]

\[
\begin{align*}
\forall \text{name z. } P \text{ z (Var name)} \\
\forall \text{t1 t2 z. } [\forall \text{z. } P \text{ z t1}; \forall \text{z. } P \text{ z t2}] &\implies P \text{ z (App t1 t2)} \\
\forall \text{name t z. } [\text{name } \# \text{ z; } \forall \text{z. } P \text{ z t}] &\implies P \text{ z Lam [name].t} \\
\hline \\
\hline \\
\hline \\
\end{align*}
\]

P \text{ t} \\

P \text{ z t}
We prove $P c t$ by induction on $t$. 
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$. 
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.

I.e., we have to show $P \ c \ (\pi \cdot (\lambda x. t))$
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.
I.e., we have to show $P \ c \ \lambda(\pi \cdot x).(\pi \cdot t)$
We prove $\forall \pi \ c. \ P c \ (\pi \cdot t)$ by induction on $t$.

I.e., we have to show $P c \ \lambda(\pi \cdot x). (\pi \cdot t)$

We have $\forall \pi \ c. \ P c \ (\pi \cdot t)$ by induction.
We prove $\forall \pi c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.
I.e., we have to show $P \ c \ \lambda(\pi \cdot x). (\pi \cdot t)$
We have $\forall \pi c. \ P \ c \ (\pi \cdot t)$ by induction.
Our weaker precondition says that:

$$\forall x \ t \ c. \ x \not\in c \land (\forall c. \ P \ c \ t) \Rightarrow \ P \ c \ (\lambda x. t)$$
• We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.

• I.e., we have to show $P \ c \ \lambda (\pi \cdot x).(\pi \cdot t)$

• We have $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction.

• Our weaker precondition says that:

$$\forall x \ t \ c. \ x \ # \ c \ \land \ (\forall c. \ P \ c \ t) \ \Rightarrow \ P \ c \ (\lambda x. t)$$

• We choose a fresh $y$ such that $y \ # \ (\pi \cdot x, \pi \cdot t, \ c)$. 
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.

I.e., we have to show $P \ c \ \lambda(\pi \cdot x). (\pi \cdot t)$

We have $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction.

Our weaker precondition says that:

$$\forall x \ t \ c. \ x \not\equiv c \land (\forall c. \ P \ c \ t) \Rightarrow P \ c \ (\lambda x. t)$$

We choose a fresh $y$ such that $y \not\equiv (\pi \cdot x, \pi \cdot t, c)$.

Now we can use $\forall c. \ P \ c \ (((y \ \pi \cdot x) :: \pi) \cdot t)$
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.

I.e., we have to show $P \ c \ \lambda (\pi \cdot x) . (\pi \cdot t)$

We have $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction.

Our weaker precondition says that:

$$\forall x \ t \ c. \ x \not\equiv c \land (\forall c. \ P \ c \ t) \Rightarrow P \ c \ (\lambda x . t)$$

We choose a fresh $y$ such that $y \not\equiv (\pi \cdot x, \pi \cdot t, c)$.

Now we can use $\forall c. \ P \ c \ ((y \ \pi \cdot x) \cdot \pi \cdot t)$
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.

I.e., we have to show $P \ c \ \lambda (\pi \cdot x). (\pi \cdot t)$

We have $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction.

Our weaker precondition says that:

$$\forall x \ t \ c. \ x \ # \ c \land (\forall c. \ P \ c \ t) \Rightarrow P \ c \ (\lambda x. t)$$

We choose a fresh $y$ such that $y \ # \ (\pi \cdot x, \pi \cdot t, c)$.

Now we can use $\forall c. \ P \ c \ ((y \ \pi \cdot x) \cdot \pi \cdot t)$

$$P \ c \ \lambda y. ((y \ \pi \cdot x) \cdot \pi \cdot t)$$
We prove $\forall \pi \ c. \ P c (\pi \cdot t)$ by induction on $t$.
I.e., we have to show $P c \lambda (\pi \cdot x). (\pi \cdot t)$
We have $\forall \pi \ c. \ P c (\pi \cdot t)$ by induction.
Our weaker precondition says that:

$$\forall x \ t \ c. \ x \not\equiv c \land (\forall c. \ P c \ t) \Rightarrow P c (\lambda x. t)$$

We choose a fresh $y$ such that $y \not\equiv (\pi \cdot x, \pi \cdot t, c)$.
Now we can use $\forall c. \ P c ((y \pi \cdot x)\cdot \pi \cdot t)$

$$P c \lambda y. ((y \pi \cdot x)\cdot \pi \cdot t)$$

However

$$\lambda y. ((y \pi \cdot x)\cdot \pi \cdot t) = \lambda (\pi \cdot x). (\pi \cdot t)$$
We prove $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction on $t$.

I.e., we have to show $P \ c \ \lambda (\pi \cdot x). (\pi \cdot t)$

We have $\forall \pi \ c. \ P \ c \ (\pi \cdot t)$ by induction.

Our weaker precondition says that:

$$\forall x \ t \ c. \ x \not\# \ c \land (\forall c. \ P \ c \ t) \Rightarrow P \ c \ (\lambda x. t)$$

We choose a fresh $y$ such that $y \not\# (\pi \cdot x, \pi \cdot t, c)$.

Now we can use $\forall c. \ P \ c \ ((y \ \pi \cdot x) \cdot \pi \cdot t)$

$$P \ c \ \lambda y. ((y \ \pi \cdot x) \cdot \pi \cdot t)$$

However

$$\lambda y. ((y \ \pi \cdot x) \cdot \pi \cdot t) = \lambda (\pi \cdot x). (\pi \cdot t)$$

Therefore $P \ c \ \lambda (\pi \cdot x). (\pi \cdot t)$ and we are done.
HOL Logic

- HOL consists of a small core of axioms, e.g.

\[
\begin{align*}
t &= t & P &= \text{True} \lor P = \text{False} \\
\forall x. f x &= g x & P \rightarrow Q & P & Q \rightarrow Q & P \rightarrow Q
\end{align*}
\]

- Definitions include

\[
\begin{align*}
\text{True} & \equiv (\lambda x. x) = (\lambda x. x) \\
\text{All } P & \equiv P = (\lambda x. \text{True}) \\
P \land Q & \equiv \forall R. (P \rightarrow Q \rightarrow R) \rightarrow R
\end{align*}
\]
Strong induction principles are crucial to make the formal reasoning about lambda-terms feasible (earlier approaches not as convenient)

HOL consists of a small core of axioms; it can be extended by definitions or combination of existing theorems. For example

Point to take Home

\[ \text{thm disjI1[THEN conjI]} \]

\[ \frac{P}{(P \lor Q') \land Q} \]

\[ \frac{Q}{P \land Q' \lor Q} \]

\[ \text{thm disjI1[OF conjI]} \]