Lectures #10–11: Semantic Equivalence

CS 6371: Advanced Programming Languages

Lemma 1. The following statements are equivalent: (1) \( \langle a, \sigma \rangle \downarrow n \), (2) \( \langle a, \sigma \rangle \to^* \langle n, \sigma \rangle \), (3) \( A[a] \sigma = n \).

Lemma 2. The following statements are equivalent: (1) \( \langle b, \sigma \rangle \downarrow p \), (2) \( \langle b, \sigma \rangle \to^* \langle p, \sigma \rangle \), (3) \( B[b] \sigma = p \).

Thm 1. The following statements are equivalent: (1) \( \langle c, \sigma \rangle \downarrow \sigma' \), (2) \( \langle c, \sigma \rangle \to^* \langle \text{skip}, \sigma' \rangle \), (3) \( C[c] \sigma = \sigma' \)

Proof that (1) \(\Rightarrow\) (2). Proof is by structural induction over the derivation \( D \) of \( \langle c, \sigma \rangle \downarrow \sigma' \).

IH: If \( \langle c_0, \sigma_0 \rangle \downarrow \sigma'_0 \) has a derivation \( D_0 < D \), then \( \langle c_0, \sigma_0 \rangle \to^* \langle \text{skip}, \sigma'_0 \rangle \).

(Base) Case 1: Suppose \( D \) ends in Rule L1:

\[
D = \frac{\langle \text{skip}, \sigma \rangle \downarrow \sigma}{(L1)}
\]

Thus, \( c = \text{skip} \) and \( \sigma = \sigma' \), so we conclude that \( \langle c, \sigma \rangle \to_n \langle \text{skip}, \sigma' \rangle \).

Case 2: Suppose \( D \) ends in Rule L2:

\[
D = \frac{D_1 \quad D_2}{(L2)}
\]

\[
D_1 = \frac{\langle c_1, \sigma \rangle \downarrow \sigma_2}{(c_1; c_2, \sigma) \downarrow \sigma'}
\]

\[
D_2 = \frac{\langle c_2, \sigma_2 \rangle \downarrow \sigma'}{(c_1; c_2, \sigma) \downarrow \sigma'}
\]

By IH with \( D_0 = D_1 \) and \( D_0 = D_2 \) (respectively), we have \( \langle c_1, \sigma \rangle \to^* \langle \text{skip}, \sigma_2 \rangle \) and \( \langle c_2, \sigma_2 \rangle \to^* \langle \text{skip}, \sigma' \rangle \) (respectively).

Lemma: If \( \langle c_1, c_2, \sigma \rangle \to_n \langle \text{skip}, \sigma_2 \rangle \) then \( \langle c_1; c_2, \sigma \rangle \to^* \langle \text{skip}, \sigma_2 \rangle \).

Proof: Proof is by induction on \( n \).

Base Case: If \( n = 0 \) then \( c_1 = \text{skip} \) and \( \sigma = \sigma_2 \). Thus, \( \langle \text{skip}; \text{skip}, \sigma \rangle \to_n \langle \text{skip}; \text{skip}, \sigma_2 \rangle \).

IH2: If \( \langle c_0, \sigma_0 \rangle \to_{n-1} \langle \text{skip}; \text{skip}, \sigma'_0 \rangle \) then \( \langle c_0; c_2, \sigma_0 \rangle \to^* \langle \text{skip}; \text{skip}, \sigma'_0 \rangle \).

Inductive Case: if \( n > 0 \) then \( \langle c_1, \sigma \rangle \to_1 \langle c'_1, \sigma_3 \rangle \to_{n-1} \langle \text{skip}, \sigma_2 \rangle \). From \( \langle c_1, \sigma \rangle \to_1 \langle c'_1, \sigma_3 \rangle \), Rule S1 derives \( \langle c_1; c_2, \sigma \rangle \to_1 \langle c'_1; c_2, \sigma_3 \rangle \). Applying IH2 with \( c_0 = c'_1; c_2, \sigma_0 = \sigma_3 \), and \( \sigma_0' = \sigma_2 \), we obtain \( \langle c'_1; c_2, \sigma_3 \rangle \to^* \langle \text{skip}; c_2, \sigma_2 \rangle \). In conclusion, \( \langle c_1; c_2, \sigma \rangle \to_1 \langle c'_1; c_2, \sigma_3 \rangle \to^* \langle \text{skip}; c_2, \sigma_2 \rangle \).

From the lemma, we conclude that \( \langle c_1; c_2, \sigma \rangle \to^* \langle \text{skip}; c_2, \sigma_2 \rangle \). Rule S1 derives \( \langle \text{skip}; c_2, \sigma_2 \rangle \to_1 \langle c_2, \sigma_2 \rangle \). We already proved that \( \langle c_2, \sigma_2 \rangle \to^* \langle \text{skip}, \sigma' \rangle \), so this completes the case.

(Base) Case 3: Suppose \( D \) ends in Rule L3:

\[
D = \frac{\langle a, \sigma \rangle \downarrow i}{(L3)}
\]

\[
D = \frac{\langle a = a, \sigma \rangle \downarrow i}{\sigma[v \mapsto i]}
\]

From \( \langle a, \sigma \rangle \downarrow i \), Lemma 1 proves that \( \langle a, \sigma \rangle \to^* \langle i, \sigma \rangle \).

Lemma: If \( \langle a, \sigma \rangle \to_n \langle i, \sigma' \rangle \) then \( \langle v = a, \sigma \rangle \to^* \langle v = i, \sigma' \rangle \).

Proof: Proof is by induction on \( n \).

Base Case: If \( n = 0 \) then \( a = i \) and \( \sigma = \sigma' \). Thus, \( \langle v = a, \sigma \rangle \to_0 \langle v = i, \sigma \rangle \).

IH2: If \( \langle a_0, \sigma_0 \rangle \to_{n-1} \langle i, \sigma'_0 \rangle \) then \( \langle v = a_0, \sigma_0 \rangle \to^* \langle v = i, \sigma'_0 \rangle \).

Inductive Case: if \( n > 0 \) then \( \langle a, \sigma \rangle \to_1 \langle a_0, \sigma_2 \rangle \to_{n-1} \langle i, \sigma' \rangle \). From \( \langle a, \sigma \rangle \to_1 \langle a_0, \sigma_2 \rangle \), Rule S3 derives \( \langle v = a, \sigma \rangle \to_1 \langle v = a_0, \sigma_2 \rangle \). Applying IH2 with \( a_0 = a_2, \sigma_0 = \sigma_2, \) and \( \sigma'_0 = \sigma' \), we obtain \( \langle v = a_2, \sigma_2 \rangle \to^* \langle v = i, \sigma' \rangle \). In conclusion, \( \langle v = a, \sigma \rangle \to_1 \langle v = a_2, \sigma_2 \rangle \to^* \langle v = i, \sigma' \rangle \).

From the lemma, we conclude that \( \langle v = a, \sigma \rangle \to^* \langle v = i, \sigma \rangle \). Rule S4 derives \( \langle v = i, \sigma \rangle \to_1 \langle \text{skip}, \sigma[v \mapsto i] \rangle \), completing the case.

Cases 4–6: Left as an exercise to the reader.
Proof that (2) \(\Rightarrow\) (3). It suffices to prove that if \(\langle c, \sigma \rangle \rightarrow \langle c', \sigma' \rangle\) then \(C[c] \sigma = C[c'] \sigma'\). Proof is by structural induction over the derivation \(D\) of \(\langle c, \sigma \rangle \rightarrow \langle c', \sigma' \rangle\).

**IH:** If \(\langle c_0, \sigma_0 \rangle \rightarrow \langle c_0', \sigma_0' \rangle\) has a derivation \(D_0 \prec D\), then \(C[c_0] \sigma_0 = C[c_0'] \sigma_0'\).

**Case 1:** Suppose \(D\) ends in Rule S2:

\[ D = \frac{\langle c_1, \sigma \rangle \rightarrow \langle c_1', \sigma' \rangle}{\langle c_1; c_2, \sigma \rangle \rightarrow \langle c_1'; c_2, \sigma' \rangle} \ 	ext{(S2)} \]

Thus, \(c = c_1; c_2\) and \(c' = c_1'; c_2\). By IH with \(D_0 = D_1\), we have \(C[c_1] \sigma = C[c_1'] \sigma'\). We conclude that \(C[c_1; c_2] \sigma = C[c_2](C[c_1] \sigma) = C[c_2](C[c_1'] \sigma') = C[c_1; c_2] \sigma'\).

**Cases 2–8:** Left as an exercise to the reader. All except Case 8 (for while loops) are fairly straightforward. □

**Proof that (3) \(\Rightarrow\) (1).** Proof is by structural induction over \(c\).

**IH:** If \(C[c_0] \sigma_0 = \sigma_0'\) and \(c_0 \prec c\), then \(\langle c_0, \sigma_0 \rangle \Downarrow \sigma_0'\) is derivable.

**(Base) Case 1:** Suppose \(c = \text{skip}\). Then \(C[c] = i\), and therefore \(\sigma' = \sigma\). Rule L1 thus derives \(\langle \text{skip}, \sigma \rangle \Downarrow \sigma\).

**Case 2:** Suppose \(c = c_1; c_2\). Then \(C[c] = C[c_2](C[c_1] \sigma)\). Applying the IH with \(c_0 = c_1, \sigma_0 = \sigma, \) and \(\sigma_0' = C[c_1] \sigma\) implies that \(\langle c_1, \sigma \rangle \Downarrow C[c_1] \sigma\) has a derivation \(D_1\). Applying the IH with \(c_0 = c_2, \sigma_0 = C[c_1] \sigma, \) and \(\sigma_0' = \sigma'\) implies that \(\langle c_2, C[c_1] \sigma \rangle \Downarrow \sigma'\) has a derivation \(D_2\). We can hence derive:

\[
\begin{array}{c}
D_1 \\
\hline \\
\langle c_1, \sigma \rangle \Downarrow C[c_1] \sigma \\
\hline \\
D_2 \\
\hline \\
\langle c_1; c_2, \sigma \rangle \Downarrow \sigma' \\
\end{array}
\]

**(L2)**

**Cases 3–5:** Left as an exercise to the reader.

**Case 6:** Suppose \(c = \text{while } b \text{ do } c_1\). Then \(C[c] = \text{fix}(\Gamma)\). We will prove \(P(\text{fix}(\Gamma))\) by fixed point induction, where property \(P\) is defined by \(P(g) \equiv \forall \sigma \in g^- : \langle c, \sigma \rangle \Downarrow g(\sigma)\).

**Base Case:** \(P(\bot)\) holds vacuously.

**IH2:** Assume \(P(g)\). That is, \(\sigma_0 \in g^- \implies \langle c_0, \sigma_0 \rangle \Downarrow g(\sigma_0)\).

**IC:** Let \(\sigma \in \Gamma(g)^-\) be given.

**Case 1:** If \(B[b] \Downarrow F\) then \(\sigma' = \Gamma(g) \sigma = \sigma\). Since \(B[b] \Downarrow F\), Lemma 2 proves that \(\langle b, \sigma \rangle \Downarrow F\) has a derivation \(D_1\). Thus, we can derive:

\[
\begin{array}{c}
D_1 \\
\hline \\
\langle b, \sigma \rangle \Downarrow F \\
\langle \text{skip}, \sigma \rangle \Downarrow \sigma \\
\hline \\
\langle \text{if } b \text{ then } (c_1; c) \text{ else skip} \rangle \Downarrow \sigma \\
\end{array}
\]

**(L5)**

**Case 2:** If \(B[b] \Downarrow T\) then \(\sigma' = \Gamma(g) \sigma = g(C[c_1] \sigma)\). Since \(B[b] \Downarrow T\), Lemma 2 proves that \(\langle b, \sigma \rangle \Downarrow T\) has a derivation \(D_1\). Applying IH(1) with \(c_0 = c_1, \sigma_0 = \sigma, \) and \(\sigma_0' = C[c_1] \sigma\) implies that \(\langle c_1, \sigma \rangle \Downarrow C[c_1] \sigma\) has a derivation \(D_2\). Applying IH2 with \(c_0 = c, \sigma_0 = C[c_1] \sigma, \) and \(\sigma_0' = \sigma'\) implies that \(\langle c, C[c_1] \sigma \rangle \Downarrow g(C[c_1] \sigma) = \sigma'\) has a derivation \(D_3\). We can therefore derive:

\[
\begin{array}{c}
D_1 \\
\hline \\
\langle b, \sigma \rangle \Downarrow T \\
\hline \\
D_2 \\
\hline \\
\langle c_1; c, \sigma \rangle \Downarrow \sigma' \\
\langle c_1; c, \sigma \rangle \Downarrow \sigma' \\
\langle \text{if } b \text{ then } (c_1; c) \text{ else skip} \rangle \Downarrow \sigma' \\
\langle \text{if } b \text{ then } (c_1; c) \text{ else skip} \rangle \Downarrow \sigma' \\
\end{array}
\]

**(L6)**