Semantics and Verification of Software

Lecture 9: Axiomatic Semantics of WHILE I (Hoare Logic)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

RWTH Aachen University

noll@cs.rwth-aachen.de

http://www-i2.informatik.rwth-aachen.de/i2/svsw08/

Winter semester 2008/09

Outline

The Axiomatic Approach

Example

Obviously c satisfies the following assertions (after execution of the respective statement):

$$
\begin{array}{l}
\texttt{s:=0;} \\
\{\texttt{s} = 0\} \\
\texttt{n:=1;} \\
\{\texttt{s} = 0 \land \texttt{n} = 1\} \\
\texttt{while } \neg\texttt{(n>N) do (s:=s+n; n:=n+1)} \\
\{\texttt{s} = \sum_{i=1}^{\texttt{N}} i \land \texttt{n} > \texttt{N}\}
\end{array}
$$

where, e.g., "$\texttt{s} = 0$" means "$\sigma(\texttt{s}) = 0$ in the current state $\sigma \in \Sigma$"

Syntax of Assertion Language I

Assertions = Boolean expressions + logical variables
(to memorize previous values of program variables)

Syntactic categories:

Category	Domain	Meta variable(s)
Logical variables	$LVar$	i
Arithmetic expressions with log. var.	$LExp$	a
Assertions	$Assn$	A, B, C

Syntax of Assertion Language II

Definition (Syntax of assertions)

The syntax of *Assn* is defined by the following context-free grammar:

$$a ::= z \mid x \mid i \mid a_1\texttt{+}a_2 \mid a_1\texttt{-}a_2 \mid a_1\texttt{*}a_2 \in LExp$$
$$A ::= t \mid a_1\texttt{=}a_2 \mid a_1\texttt{>}a_2 \mid \neg A \mid A_1 \wedge A_2 \mid A_1 \vee A_2 \mid \forall i.A \in Assn$$

Abbreviations:

$$A_1 \implies A_2 := \neg A_1 \vee A_2$$
$$\exists i.A := \neg(\forall i.\neg A)$$
$$a_1 \geq a_2 := a_1\texttt{>}a_2 \vee a_1\texttt{=}a_2$$
$$\vdots$$

Outline

Semantics of *LExp*

The semantics now additionally depends on values of logical variables:

Definition 9.1 (Semantics of *LExp*)

An interpretation is an element of the set

$$Int := \{I \mid I : LVar \to \mathbb{Z}\}.$$

The value of an arithmetic expressions with logical variables is given by the functional

$$\mathfrak{L}[\![.]\!] : LExp \to (Int \to (\Sigma \to \mathbb{Z}))$$

where

$$\begin{array}{rclrcl}
\mathfrak{L}[\![z]\!]I\sigma & := & z & \mathfrak{L}[\![a_1\texttt{+}a_2]\!]I\sigma & := & \mathfrak{L}[\![a_1]\!]I\sigma + \mathfrak{L}[\![a_2]\!]I\sigma \\
\mathfrak{L}[\![x]\!]I\sigma & := & \sigma(x) & \mathfrak{L}[\![a_1\texttt{-}a_2]\!]I\sigma & := & \mathfrak{L}[\![a_1]\!]I\sigma - \mathfrak{L}[\![a_2]\!]I\sigma \\
\mathfrak{L}[\![i]\!]I\sigma & := & I(i) & \mathfrak{L}[\![a_1\texttt{*}a_2]\!]I\sigma & := & \mathfrak{L}[\![a_1]\!]I\sigma * \mathfrak{L}[\![a_2]\!]I\sigma
\end{array}$$

Semantics of $LExp$

The semantics now additionally depends on values of logical variables:

Definition 9.1 (Semantics of $LExp$)

An interpretation is an element of the set

$$Int := \{I \mid I : LVar \to \mathbb{Z}\}.$$

The value of an arithmetic expressions with logical variables is given by the functional

$$\mathfrak{L}[\![.]\!] : LExp \to (Int \to (\Sigma \to \mathbb{Z}))$$

where

$$\begin{array}{llll}
\mathfrak{L}[\![z]\!]I\sigma := z & \quad \mathfrak{L}[\![a_1 \texttt{+} a_2]\!]I\sigma := \mathfrak{L}[\![a_1]\!]I\sigma + \mathfrak{L}[\![a_2]\!]I\sigma \\
\mathfrak{L}[\![x]\!]I\sigma := \sigma(x) & \quad \mathfrak{L}[\![a_1 \texttt{-} a_2]\!]I\sigma := \mathfrak{L}[\![a_1]\!]I\sigma - \mathfrak{L}[\![a_2]\!]I\sigma \\
\mathfrak{L}[\![i]\!]I\sigma := I(i) & \quad \mathfrak{L}[\![a_1 \texttt{*} a_2]\!]I\sigma := \mathfrak{L}[\![a_1]\!]I\sigma * \mathfrak{L}[\![a_2]\!]I\sigma
\end{array}$$

Def. 4.4 (denotational semantics of arithmetic expressions) implies:

Corollary 9.2

For every $a \in AExp$ (without logical variables), $I \in Int$, and $\sigma \in \Sigma$:

$$\mathfrak{L}[\![a]\!]I\sigma = \mathfrak{A}[\![a]\!]\sigma.$$

Semantics of Assertions I

- Formalized by a satisfaction relation of the form

$$\sigma \models A$$

(where $\sigma \in \Sigma$ and $A \in Assn$)

Semantics of Assertions I

- Formalized by a satisfaction relation of the form

$$\sigma \models A$$

(where $\sigma \in \Sigma$ and $A \in Assn$)

- Non-terminating computations captured by undefined state $\perp$:

$$\Sigma_\perp := \Sigma \cup \{\perp\}$$

Semantics of Assertions I

- Formalized by a satisfaction relation of the form

$$\sigma \models A$$

(where $\sigma \in \Sigma$ and $A \in Assn$)

- Non-terminating computations captured by undefined state $\bot$:

$$\Sigma_\bot := \Sigma \cup \{\bot\}$$

- Modification of interpretations (in analogy to program states):

$$I[i \mapsto z](j) := \begin{cases} z & \text{if } j = i \\ I(j) & \text{otherwise} \end{cases}$$

Semantics of Assertions II

Reminder:

$A ::= t \mid a_1 = a_2 \mid a_1 > a_2 \mid \neg A \mid A_1 \wedge A_2 \mid A_1 \vee A_2 \mid \forall i.A \in Assn$

Definition 9.3 (Semantics of assertions)

Let $A \in Assn$, $\sigma \in \Sigma_\perp$, and $I \in Int$. The relation "σ satisfies A in I" (notation: $\sigma \models^I A$) is inductively defined by:

$$\begin{array}{rll}
\sigma & \models^I \mathsf{true} & \\
\sigma & \models^I a_1 = a_2 & \text{if } \mathfrak{L}[\![a_1]\!]I\sigma = \mathfrak{L}[\![a_2]\!]I\sigma \\
\sigma & \models^I a_1 > a_2 & \text{if } \mathfrak{L}[\![a_1]\!]I\sigma > \mathfrak{L}[\![a_2]\!]I\sigma \\
\sigma & \models^I \neg A & \text{if not } \sigma \models^I A \\
\sigma & \models^I A_1 \wedge A_2 & \text{if } \sigma \models^I A_1 \text{ and } \sigma \models^I A_2 \\
\sigma & \models^I A_1 \vee A_2 & \text{if } \sigma \models^I A_1 \text{ or } \sigma \models^I A_2 \\
\sigma & \models^I \forall i.A & \text{if } \sigma \models^{I[i \mapsto z]} A \text{ for every } z \in \mathbb{Z} \\
\perp & \models^I A &
\end{array}$$

Furthermore "σ satisfies A" ($\sigma \models A$) if $\sigma \models^I A$ for every interpretation $I \in Int$, and A is called valid ($\models A$) if $\sigma \models A$ for every state $\sigma \in \Sigma$.

Semantics of Assertions III

Example 9.4

The following assertion expresses that, in the current state $\sigma \in \Sigma$, $\sigma(y)$ is the greatest divisor of $\sigma(x)$:

$$(\exists i. i > 1 \wedge i * y = x) \wedge \forall j. \forall k. (j > 1 \wedge j * k = x \implies k \leq y)$$

Semantics of Assertions III

Example 9.4

The following assertion expresses that, in the current state $\sigma \in \Sigma$, $\sigma(y)$ is the greatest divisor of $\sigma(x)$:

$$(\exists i.i > 1 \land i{*}y = x) \land \forall j.\forall k.(j > 1 \land j{*}k = x \implies k \leq y)$$

In analogy to Corollary 9.2, Def. 4.5 (denotational semantics of Boolean expressions) yields:

Corollary 9.5

For every $b \in BExp$ (without logical variables), $I \in Int$, and $\sigma \in \Sigma$:

$$\sigma \models^I b \iff \mathfrak{B}[\![b]\!]\sigma = \mathsf{true}.$$

Semantics of Assertions IV

Definition 9.6 (Extension)

Let $A \in Assn$ and $I \in Int$. The extension of A with respect to I is given by

$$A^I := \{\sigma \in \Sigma_\perp \mid \sigma \models^I A\}.$$

Note that, for every $A \in Assn$ and $I \in Int$, $\perp \in A^I$.

Semantics of Assertions IV

Definition 9.6 (Extension)

Let $A \in Assn$ and $I \in Int$. The extension of A with respect to I is given by
$$A^I := \{\sigma \in \Sigma_\perp \mid \sigma \models^I A\}.$$

Note that, for every $A \in Assn$ and $I \in Int$, $\perp \in A^I$.

Example 9.7

For $A := (\exists i.i{*}i = x)$ and every $I \in Int$,
$$A^I = \{\perp\} \cup \{\sigma \in \Sigma \mid \sigma(x) \in \{0, 1, 4, 9, \ldots\}\}$$

Outline

Partial Correctness Properties

Definition 9.8 (Partial correctness properties)

Let $A, B \in Assn$ and $c \in Cmd$.

- An expression of the form $\{A\}\,c\,\{B\}$ is called a partial correctness property with precondition A and postcondition B.

Definition 9.8 (Partial correctness properties)

Let $A, B \in Assn$ and $c \in Cmd$.

- An expression of the form $\{A\}\,c\,\{B\}$ is called a partial correctness property with precondition A and postcondition B.
- Given $\sigma \in \Sigma_\perp$ and $I \in Int$, we let

$$\sigma \models^I \{A\}\,c\,\{B\}$$

  if $\sigma \models^I A$ implies $\mathfrak{C}[\![c]\!]\sigma \models^I B$
  (or equivalently: $\sigma \in A^I \implies \mathfrak{C}[\![c]\!]\sigma \in B^I$).

Partial Correctness Properties

Definition 9.8 (Partial correctness properties)

Let $A, B \in Assn$ and $c \in Cmd$.

- An expression of the form $\{A\}\,c\,\{B\}$ is called a partial correctness property with precondition A and postcondition B.
- Given $\sigma \in \Sigma_\perp$ and $I \in Int$, we let

$$\sigma \models^I \{A\}\,c\,\{B\}$$

  if $\sigma \models^I A$ implies $\mathfrak{C}[\![c]\!]\sigma \models^I B$
  (or equivalently: $\sigma \in A^I \implies \mathfrak{C}[\![c]\!]\sigma \in B^I$).
- $\{A\}\,c\,\{B\}$ is called valid in I (notation: $\models^I \{A\}\,c\,\{B\}$) if $\sigma \models^I \{A\}\,c\,\{B\}$ for every $\sigma \in \Sigma_\perp$ (or equivalently: $\mathfrak{C}[\![c]\!]A^I \subseteq B^I$).

Partial Correctness Properties

Definition 9.8 (Partial correctness properties)

Let $A, B \in Assn$ and $c \in Cmd$.

- An expression of the form $\{A\}\,c\,\{B\}$ is called a partial correctness property with precondition A and postcondition B.
- Given $\sigma \in \Sigma_\perp$ and $I \in Int$, we let

$$\sigma \models^I \{A\}\,c\,\{B\}$$

  if $\sigma \models^I A$ implies $\mathfrak{C}[\![c]\!]\sigma \models^I B$
  (or equivalently: $\sigma \in A^I \implies \mathfrak{C}[\![c]\!]\sigma \in B^I$).
- $\{A\}\,c\,\{B\}$ is called valid in I (notation: $\models^I \{A\}\,c\,\{B\}$) if $\sigma \models^I \{A\}\,c\,\{B\}$ for every $\sigma \in \Sigma_\perp$ (or equivalently: $\mathfrak{C}[\![c]\!]A^I \subseteq B^I$).
- $\{A\}\,c\,\{B\}$ is called valid (notation: $\models \{A\}\,c\,\{B\}$) if $\models^I \{A\}\,c\,\{B\}$ for every $I \in Int$.

Outline

A Valid Partial Correctness Property

Example 9.9

- Let $\mathtt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:

$$\models \{i \leq \mathtt{x}\}\, \mathtt{x := x+1}\, \{i < \mathtt{x}\}$$

A Valid Partial Correctness Property

Example 9.9

- Let $\mathtt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:
$$\models \{i \leq \mathtt{x}\}\, \mathtt{x\ :=\ x+1}\, \{i < \mathtt{x}\}$$
- According to Def. 9.8, this is equivalent to
$$\sigma \models^I \{i \leq \mathtt{x}\}\, \mathtt{x\ :=\ x+1}\, \{i < \mathtt{x}\}$$
for every $\sigma \in \Sigma_\perp$ and $I \in \mathit{Int}$

A Valid Partial Correctness Property

Example 9.9

- Let $\texttt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:
$$\models \{i \leq \texttt{x}\}\,\texttt{x := x+1}\,\{i < \texttt{x}\}$$
- According to Def. 9.8, this is equivalent to
$$\sigma \models^I \{i \leq \texttt{x}\}\,\texttt{x := x+1}\,\{i < \texttt{x}\}$$
for every $\sigma \in \Sigma_\perp$ and $I \in \mathit{Int}$
- For $\sigma = \perp$ this is trivial. So let $\sigma \in \Sigma$:
$$\sigma \models^I (i \leq \texttt{x})$$

A Valid Partial Correctness Property

Example 9.9

- Let $\texttt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:

$$\models \{i \leq \texttt{x}\}\, \texttt{x := x+1}\, \{i < \texttt{x}\}$$

- According to Def. 9.8, this is equivalent to

$$\sigma \models^{I} \{i \leq \texttt{x}\}\, \texttt{x := x+1}\, \{i < \texttt{x}\}$$

  for every $\sigma \in \Sigma_\perp$ and $I \in \mathit{Int}$

- For $\sigma = \perp$ this is trivial. So let $\sigma \in \Sigma$:

$$\begin{aligned} & \sigma \models^{I} (i \leq \texttt{x}) \\ \Longrightarrow\ & \mathfrak{L}[\![i]\!]I\sigma \leq \mathfrak{L}[\![\texttt{x}]\!]I\sigma \quad \text{(Def. 9.3)} \end{aligned}$$

A Valid Partial Correctness Property

Example 9.9

- Let $\mathtt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:
$$\models \{i \leq \mathtt{x}\}\, \mathtt{x\ :=\ x+1}\, \{i < \mathtt{x}\}$$
- According to Def. 9.8, this is equivalent to
$$\sigma \models^I \{i \leq \mathtt{x}\}\, \mathtt{x\ :=\ x+1}\, \{i < \mathtt{x}\}$$
for every $\sigma \in \Sigma_\perp$ and $I \in \mathit{Int}$
- For $\sigma = \perp$ this is trivial. So let $\sigma \in \Sigma$:
$$\begin{aligned} & \sigma \models^I (i \leq \mathtt{x}) \\ \Longrightarrow\ & \mathfrak{L}[\![i]\!]I\sigma \leq \mathfrak{L}[\![\mathtt{x}]\!]I\sigma \quad \text{(Def. 9.3)} \\ \Longrightarrow\ & I(i) \leq \sigma(\mathtt{x}) \quad \text{(Def. 9.1)} \end{aligned}$$

A Valid Partial Correctness Property

Example 9.9

- Let $\mathtt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:
$$\models \{i \leq \mathtt{x}\}\,\mathtt{x\ :=\ x+1}\,\{i < \mathtt{x}\}$$
- According to Def. 9.8, this is equivalent to
$$\sigma \models^{I} \{i \leq \mathtt{x}\}\,\mathtt{x\ :=\ x+1}\,\{i < \mathtt{x}\}$$
for every $\sigma \in \Sigma_{\perp}$ and $I \in \mathit{Int}$
- For $\sigma = \perp$ this is trivial. So let $\sigma \in \Sigma$:
$$\begin{aligned} & \sigma \models^{I} (i \leq \mathtt{x}) \\ \Longrightarrow\ & \mathfrak{L}[\![i]\!]I\sigma \leq \mathfrak{L}[\![\mathtt{x}]\!]I\sigma \quad \text{(Def. 9.3)} \\ \Longrightarrow\ & I(i) \leq \sigma(\mathtt{x}) \quad \text{(Def. 9.1)} \\ \Longrightarrow\ & I(i) < \sigma(\mathtt{x}) + 1 \\ & \qquad = (\mathfrak{C}[\![\mathtt{x\ :=\ x+1}]\!]\sigma)(\mathtt{x}) \end{aligned}$$

A Valid Partial Correctness Property

Example 9.9

- Let $\texttt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:
$$\models \{i \leq \texttt{x}\}\, \texttt{x := x+1}\, \{i < \texttt{x}\}$$
- According to Def. 9.8, this is equivalent to
$$\sigma \models^I \{i \leq \texttt{x}\}\, \texttt{x := x+1}\, \{i < \texttt{x}\}$$
for every $\sigma \in \Sigma_\perp$ and $I \in \mathit{Int}$
- For $\sigma = \perp$ this is trivial. So let $\sigma \in \Sigma$:
$$\begin{aligned} & \sigma \models^I (i \leq \texttt{x}) \\ \Longrightarrow\ & \mathfrak{L}[\![i]\!]I\sigma \leq \mathfrak{L}[\![\texttt{x}]\!]I\sigma \quad \text{(Def. 9.3)} \\ \Longrightarrow\ & I(i) \leq \sigma(\texttt{x}) \quad \text{(Def. 9.1)} \\ \Longrightarrow\ & I(i) < \sigma(\texttt{x}) + 1 \\ & = (\mathfrak{C}[\![\texttt{x := x+1}]\!]\sigma)(\texttt{x}) \\ \Longrightarrow\ & \mathfrak{C}[\![\texttt{x := x+1}]\!]\sigma \models^I (i < \texttt{x}) \end{aligned}$$

A Valid Partial Correctness Property

Example 9.9

- Let $\mathtt{x} \in \mathit{Var}$ and $i \in \mathit{LVar}$. We have to show:
$$\models \{i \leq \mathtt{x}\}\, \mathtt{x\ :=\ x+1}\, \{i < \mathtt{x}\}$$
- According to Def. 9.8, this is equivalent to
$$\sigma \models^I \{i \leq \mathtt{x}\}\, \mathtt{x\ :=\ x+1}\, \{i < \mathtt{x}\}$$
for every $\sigma \in \Sigma_\perp$ and $I \in \mathit{Int}$
- For $\sigma = \perp$ this is trivial. So let $\sigma \in \Sigma$:
$$\begin{aligned} & \sigma \models^I (i \leq \mathtt{x}) \\ \Longrightarrow\ & \mathfrak{L}[\![i]\!]I\sigma \leq \mathfrak{L}[\![\mathtt{x}]\!]I\sigma \quad \text{(Def. 9.3)} \\ \Longrightarrow\ & I(i) \leq \sigma(\mathtt{x}) \quad \text{(Def. 9.1)} \\ \Longrightarrow\ & I(i) < \sigma(\mathtt{x}) + 1 \\ & \qquad\quad = (\mathfrak{C}[\![\mathtt{x\ :=\ x+1}]\!]\sigma)(\mathtt{x}) \\ \Longrightarrow\ & \mathfrak{C}[\![\mathtt{x\ :=\ x+1}]\!]\sigma \models^I (i < \mathtt{x}) \\ \Longrightarrow\ & \text{claim} \end{aligned}$$

Outline

Goal: syntactic derivation of valid partial correctness properties

Definition 9.10 (Hoare Logic)

The Hoare rules are given by

$$\text{(skip)}\ \frac{}{\{A\}\,\texttt{skip}\,\{A\}} \qquad \text{(asgn)}\ \frac{}{\{A[x \mapsto a]\}\,x\,\texttt{:=}a\,\{A\}}$$

$$\text{(seq)}\ \frac{\{A\}\,c_1\,\{C\}\ \{C\}\,c_2\,\{B\}}{\{A\}\,c_1\,\texttt{;}c_2\,\{B\}} \qquad \text{(if)}\ \frac{\{A \wedge b\}\,c_1\,\{B\}\ \{A \wedge \neg b\}\,c_2\,\{B\}}{\{A\}\,\texttt{if}\ b\ \texttt{then}\ c_1\ \texttt{else}\ c_2\,\{B\}}$$

$$\text{(while)}\ \frac{\{A \wedge b\}\,c\,\{A\}}{\{A\}\,\texttt{while}\ b\ \texttt{do}\ c\,\{A \wedge \neg b\}}$$

$$\text{(cons)}\ \frac{\models (A \implies A')\ \{A'\}\,c\,\{B'\}\ \models (B' \implies B)}{\{A\}\,c\,\{B\}}$$

A partial correctness property is provable (notation: $\vdash \{A\}\,c\,\{B\}$) if it is derivable by the Hoare rules. In case of (while), A is called a (loop) invariant.

Here $A[x \mapsto a]$ denotes the syntactic replacement of every occurrence of x by a in A.

Example 9.11

Proof of $\{A\}$ `y:=1;`c $\{B\}$ where

$$c := (\texttt{while}\ \neg\texttt{(x=1)}\ \texttt{do}\ \texttt{(y:=y*x; x:=x-1)})$$
$$A := (\texttt{x} = i)$$
$$B := (\texttt{y} = i!)$$

(on the board)

Example 9.11

Proof of $\{A\}\,\texttt{y:=1;}c\,\{B\}$ where

$$
\begin{aligned}
c &:= (\texttt{while } \neg\texttt{(x=1) do (y:=y*x; x:=x-1)}) \\
A &:= (\texttt{x} = i) \\
B &:= (\texttt{y} = i!)
\end{aligned}
$$

(on the board)

Structure of the proof:

$$
\text{(seq)}\ \frac{\text{(cons)}\ \frac{\frac{}{4}\ \text{(asgn)}\ \frac{}{5}\ \ \frac{}{6}}{2}\quad \text{(cons)}\ \frac{\frac{}{7}\ \text{(while)}\ \frac{\text{(cons)}\ \frac{\frac{}{11}\ \text{(seq)}\ \frac{\text{(asgn)}\ \frac{}{14}\ \text{(asgn)}\ \frac{}{15}}{12}\ \ \frac{}{13}}{10}}{8}\ \ \frac{}{9}}{3}}{1}
$$

Example 9.11 (continued)

Here the single propositions are given by:

0. $C := (\mathtt{x} > 0 \implies \mathtt{y} * \mathtt{x}! = i! \land i \geq \mathtt{x})$
1. $\{A\}\,\mathtt{y\ :=\ 1;}c\,\{B\}$
2. $\{A\}\,\mathtt{y\ :=\ 1}\,\{C\}$
3. $\{C\}\,c\,\{B\}$
4. $\models (A \implies C[\mathtt{y} \mapsto 1])$
5. $\{C[\mathtt{y} \mapsto 1]\}\,\mathtt{y\ :=\ 1}\,\{C\}$
6. $\models (C \implies C)$
7. $\models (C \implies C)$
8. $\{C\}\,c\,\{\neg(\neg(\mathtt{x\ =\ 1})) \land C\}$
9. $\models (\neg(\neg(\mathtt{x\ =\ 1})) \land C \implies B)$
10. $\{\neg(\mathtt{x\ =\ 1}) \land C\}\,\mathtt{y\ :=\ y*x;\ \ x\ :=\ x-1}\,\{C\}$
11. $\models (\neg(\mathtt{x\ =\ 1}) \land C \implies C[\mathtt{x} \mapsto \mathtt{x-1}, \mathtt{y} \mapsto \mathtt{y*x}])$
12. $\{C[\mathtt{x} \mapsto \mathtt{x-1}, \mathtt{y} \mapsto \mathtt{y*x}]\}\,\mathtt{y\ :=\ y*x;\ \ x\ :=\ x-1}\,\{C\}$
13. $\models (C \implies C)$
14. $\{C[\mathtt{x} \mapsto \mathtt{x-1}, \mathtt{y} \mapsto \mathtt{y*x}]\}\,\mathtt{y\ :=\ y*x}\,\{C[\mathtt{x} \mapsto \mathtt{x-1}]\}$
15. $\{C[\mathtt{x} \mapsto \mathtt{x-1}]\}\,\mathtt{x\ :=\ x-1}\,\{C\}$