



Semantics and Verification of Software

Summer Semester 2019

Lecture 12: Axiomatic Semantics of WHILE IV
(Total Correctness & Axiomatic Equivalence)

Thomas Noll

Software Modeling and Verification Group

RWTH Aachen University

<https://moves.rwth-aachen.de/teaching/ss-19/sv-sw/>

Recap: Completeness & Total Correctness

Outline of Lecture 12

Recap: Completeness & Total Correctness

Soundness and Completeness of Hoare Logic for Total Correctness

Axiomatic Equivalence

Characteristic Assertions

Partial vs. Total Equivalence

Recap: Completeness & Total Correctness

Relative Completeness of Hoare Logic

Theorem (Cook's Completeness Theorem)

Hoare Logic is *relatively complete*, i.e., for every partial correctness property $\{A\} c \{B\}$:

$$\models \{A\} c \{B\} \Rightarrow \vdash \{A\} c \{B\}.$$



Stephen A. Cook (* 1939)

Thus: if we know that a partial correctness property is valid, then we know that there is a corresponding proof.

The proof uses the following concept: assume that, e.g., $\{A\} c_1 ; c_2 \{B\}$ has to be derived. This requires an *intermediate assertion* $C \in Assn$ such that $\{A\} c_1 \{C\}$ and $\{C\} c_2 \{B\}$. How to find it?

Recap: Completeness & Total Correctness

Weakest Liberal Preconditions

Definition (Weakest liberal precondition)

Given $c \in \text{Cmd}$ and $S \subseteq \Sigma$, the **weakest (liberal) precondition** of S with respect to c collects all states σ such that running c in σ does not terminate or yields a state in S :

$$\text{wlp}[c]S := \{\sigma \in \Sigma \mid \mathcal{E}[c]\sigma \in S \cup \{\perp\}\}.$$

Corollary

For every $c \in \text{Cmd}$, $A, B \in \text{Assn}$, and $I \in \text{Int}$:

1. $\models' \{A\} c \{B\} \iff A' \subseteq \text{wlp}[c]B'$
2. If $A_0 \in \text{Assn}$ such that $A_0^I = \text{wlp}[c]B'$ for every $I \in \text{Int}$, then $\models \{A\} c \{B\} \iff \models (A \Rightarrow A_0)$

Remarks:

- Corollary 11.5 justifies the notion of **weakest** precondition: it is entailed by every precondition A that makes $\{A\} c \{B\}$ valid.
- In the following, we do not distinguish between sets of program states (such as S or A') and predicates on program states (such as $\mathfrak{B}[b]$).

Recap: Completeness & Total Correctness

Proving Total Correctness

Definition (Hoare Logic for total correctness)

The **Hoare rules for total correctness** are given by (where $i \in LVar$)

$$\frac{}{\text{(skip)} \quad \{A\} \text{ skip } \{\Downarrow A\}}$$

$$\frac{}{\text{(asgn)} \quad \{A[x \mapsto a]\} x := a \{\Downarrow A\}}$$

$$\frac{\{A\} c_1 \{\Downarrow C\} \quad \{C\} c_2 \{\Downarrow B\}}{\text{(seq)} \quad \{A\} c_1 ; c_2 \{\Downarrow B\}}$$

$$\frac{\{A \wedge b\} c_1 \{\Downarrow B\} \quad \{A \wedge \neg b\} c_2 \{\Downarrow B\}}{\text{(if)} \quad \{A\} \text{ if } b \text{ then } c_1 \text{ else } c_2 \text{ end } \{\Downarrow B\}}$$

$$\frac{\models (i \geq 0 \wedge A(i+1) \Rightarrow b) \quad \{i \geq 0 \wedge A(i+1)\} c \{\Downarrow A(i)\} \quad \models (A(0) \Rightarrow \neg b)}{\text{(while)} \quad \{\exists i. i \geq 0 \wedge A(i)\} \text{ while } b \text{ do } c \text{ end } \{\Downarrow A(0)\}}$$

$$\frac{\models (A \Rightarrow A') \quad \{A'\} c \{\Downarrow B'\} \quad \models (B' \Rightarrow B)}{\text{(cons)} \quad \{A\} c \{\Downarrow B\}}$$

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

Soundness and Completeness of Hoare Logic for Total Correctness

Outline of Lecture 12

Recap: Completeness & Total Correctness

Soundness and Completeness of Hoare Logic for Total Correctness

Axiomatic Equivalence

Characteristic Assertions

Partial vs. Total Equivalence

Soundness and Completeness of Hoare Logic for Total Correctness

Soundness

In analogy to Theorem 10.4 we can show that the Hoare Logic for total correctness properties is also sound:

Theorem 12.1 (Soundness)

For every total correctness property $\{A\} c \{\Downarrow B\}$,

$$\vdash \{A\} c \{\Downarrow B\} \quad \Rightarrow \quad \models \{A\} c \{\Downarrow B\}.$$

Soundness and Completeness of Hoare Logic for Total Correctness

Soundness

In analogy to Theorem 10.4 we can show that the Hoare Logic for total correctness properties is also sound:

Theorem 12.1 (Soundness)

For every total correctness property $\{A\} c \{\Downarrow B\}$,

$$\vdash \{A\} c \{\Downarrow B\} \Rightarrow \models \{A\} c \{\Downarrow B\}.$$

Proof.

Again by structural induction over the derivation tree of $\vdash \{A\} c \{\Downarrow B\}$. Here we only consider the (while) case:

$$\frac{\vdash (i \geq 0 \wedge A(i+1) \Rightarrow b) \quad \{i \geq 0 \wedge A(i+1)\} c \{\Downarrow A(i)\} \quad \vdash (A(0) \Rightarrow \neg b)}{\text{(while)} \quad \{\exists i. i \geq 0 \wedge A(i)\} \text{ while } b \text{ do } c \text{ end } \{\Downarrow A(0)\}}$$

(on the board)



Soundness and Completeness of Hoare Logic for Total Correctness

Relative Completeness

Also the counterpart to Cook's Completeness Theorem 11.3 applies:

Theorem 12.2 (Completeness)

The Hoare Logic for total correctness properties is *relatively complete*, i.e., for every $\{A\} c \{\Downarrow B\}$:

$$\models \{A\} c \{\Downarrow B\} \Rightarrow \vdash \{A\} c \{\Downarrow B\}.$$

Proof.

again using weakest preconditions (see following slides)



Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions I

Definition 12.3 (Weakest precondition)

Given $c \in \text{Cmd}$ and $S \subseteq \Sigma$, the **weakest precondition** of S with respect to c collects all states σ such that executing c in σ terminates and yields a state in S :

$$wp[c]S := \{\sigma \in \Sigma \mid \mathcal{C}[c]\sigma \in S\}.$$

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions I

Definition 12.3 (Weakest precondition)

Given $c \in \text{Cmd}$ and $S \subseteq \Sigma$, the **weakest precondition** of S with respect to c collects all states σ such that executing c in σ terminates and yields a state in S :

$$wp[c]S := \{\sigma \in \Sigma \mid \mathcal{C}[c]\sigma \in S\}.$$

Corollary 12.4

For every $c \in \text{Cmd}$, $A, B \in \text{Assn}$, and $I \in \text{Int}$:

1. $\models^I \{A\} c \{\Downarrow B\} \iff A^I \subseteq wp[c]B^I$
2. If $A_0 \in \text{Assn}$ such that $A_0^I = wp[c]B^I$ for every $I \in \text{Int}$, then

$$\models \{A\} c \{\Downarrow B\} \iff \models (A \Rightarrow A_0)$$

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions II

Lemma 12.5 (Weakest precondition transformer)

Weakest preconditions $wp[\cdot]. : Cmd \times 2^\Sigma \rightarrow 2^\Sigma$ can be computed as follows:

$$wp[\text{skip}] S = S$$

$$wp[x := a] S = \{\sigma \in \Sigma \mid \sigma[x \mapsto \mathcal{A}[a]\sigma] \in S\}$$

$$wp[c_1 ; c_2] S = wp[c_1](wp[c_2] S)$$

$$wp[\text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end}] S = (\mathcal{B}[b] \cap wp[c_1] S) \cup (\mathcal{B}[\neg b] \cap wp[c_2] S)$$

$$wp[\text{while } b \text{ do } c \text{ end}] S = \text{fix}(\Psi)$$

where $\text{fix}(\Psi)$ denotes the least fixpoint (w.r.t. $(2^\Sigma, \subseteq)$) of

$$\Psi : 2^\Sigma \rightarrow 2^\Sigma : T \mapsto (\mathcal{B}[b] \cap wlp[c] T) \cup (\mathcal{B}[\neg b] \cap S)$$

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions II

Lemma 12.5 (Weakest precondition transformer)

Weakest preconditions $wp[\cdot]. : Cmd \times 2^\Sigma \rightarrow 2^\Sigma$ can be computed as follows:

$$\begin{aligned}wp[\text{skip}] S &= S \\wp[x := a] S &= \{\sigma \in \Sigma \mid \sigma[x \mapsto \mathcal{A}[a]\sigma] \in S\} \\wp[c_1 ; c_2] S &= wp[c_1](wp[c_2] S) \\wp[\text{if } b \text{ then } c_1 \text{ else } c_2 \text{ end}] S &= (\mathcal{B}[b] \cap wp[c_1] S) \cup (\mathcal{B}[\neg b] \cap wp[c_2] S) \\wp[\text{while } b \text{ do } c \text{ end}] S &= \text{fix}(\Psi)\end{aligned}$$

where $\text{fix}(\Psi)$ denotes the least fixpoint (w.r.t. $(2^\Sigma, \subseteq)$) of

$$\Psi : 2^\Sigma \rightarrow 2^\Sigma : T \mapsto (\mathcal{B}[b] \cap wlp[c] T) \cup (\mathcal{B}[\neg b] \cap S)$$

Proof.

omitted □

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions III

Example 12.6 (cf. Example 11.7)

Using Lemma 12.5, we want to determine the weakest precondition for

$$\{?\} \underbrace{\text{while } x \neq 0 \wedge x \neq 1 \text{ do } \overbrace{x := x-2}^{c_0} \text{ end}}_c \{x = 1\}$$

i.e., $wp[[c]]S$ for $S := \mathfrak{B}[[x = 1]] = \{\sigma \in \Sigma \mid \sigma(x) = 1\}$.

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions III

Example 12.6 (cf. Example 11.7)

Using Lemma 12.5, we want to determine the weakest precondition for

$$\{?\} \underbrace{\text{while } x \neq 0 \wedge x \neq 1 \text{ do } \overbrace{x := x-2}^{c_0} \text{ end}}_c \{x = 1\}$$

i.e., $wp[[c]] S$ for $S := \mathfrak{B}[[x = 1]] = \{\sigma \in \Sigma \mid \sigma(x) = 1\}$.

- $wp[[c]] S = \text{fix}(\Psi)$ for $\Psi(T) = (\mathfrak{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]] T) \cup \underbrace{(\mathfrak{B}[[x \in \{0, 1\}]] \cap S)}_{=S}$

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions III

Example 12.6 (cf. Example 11.7)

Using Lemma 12.5, we want to determine the weakest precondition for

$$\{?\} \underbrace{\text{while } x \neq 0 \wedge x \neq 1 \text{ do } \overbrace{x := x-2}^{c_0} \text{ end}}_c \{x = 1\}$$

i.e., $wp[[c]] S$ for $S := \mathfrak{B}[[x = 1]] = \{\sigma \in \Sigma \mid \sigma(x) = 1\}$.

- $wp[[c]] S = \text{fix}(\Psi)$ for $\Psi(T) = (\mathfrak{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]] T) \cup \underbrace{(\mathfrak{B}[[x \in \{0, 1\}]] \cap S)}_{=S}$
- $wp[[c_0]] T = \{\sigma \in \Sigma \mid \sigma[x \mapsto \sigma(x) - 2] \in T\}$

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions III

Example 12.6 (cf. Example 11.7)

Using Lemma 12.5, we want to determine the weakest precondition for

$$\{?\} \underbrace{\text{while } x \neq 0 \wedge x \neq 1 \text{ do } \overbrace{x := x-2}^{c_0} \text{ end}}_c \{x = 1\}$$

i.e., $wp[[c]] S$ for $S := \mathcal{B}[[x = 1]] = \{\sigma \in \Sigma \mid \sigma(x) = 1\}$.

- $wp[[c]] S = \text{fix}(\Psi)$ for $\Psi(T) = (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]] T) \cup \underbrace{(\mathcal{B}[[x \in \{0, 1\}]] \cap S)}_{=S}$
- $wp[[c_0]] T = \{\sigma \in \Sigma \mid \sigma[x \mapsto \sigma(x) - 2] \in T\}$
- Fixpoint iteration (with initial value $\bigsqcup \emptyset = \emptyset$):

$$\begin{aligned}\Psi(\emptyset) &= (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]] \emptyset) \cup S = \mathcal{B}[[x = 1]] \\ \Psi^2(\emptyset) &= (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]](\mathcal{B}[[x = 1]])) \cup S = \mathcal{B}[[x \in \{1, 3\}]] \\ \Psi^3(\emptyset) &= (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]](\mathcal{B}[[x \in \{1, 3\}]])) \cup S = \mathcal{B}[[x \in \{1, 3, 5\}]] \\ &\vdots\end{aligned}$$

Soundness and Completeness of Hoare Logic for Total Correctness

Weakest Preconditions III

Example 12.6 (cf. Example 11.7)

Using Lemma 12.5, we want to determine the weakest precondition for

$$\{?\} \underbrace{\text{while } x \neq 0 \wedge x \neq 1 \text{ do } x := x-2 \text{ end}}_{c_0} \{x = 1\}$$

i.e., $wp[[c]] S$ for $S := \mathcal{B}[[x = 1]] = \{\sigma \in \Sigma \mid \sigma(x) = 1\}$.

- $wp[[c]] S = \text{fix}(\Psi)$ for $\Psi(T) = (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]] T) \cup \underbrace{(\mathcal{B}[[x \in \{0, 1\}]] \cap S)}_{=S}$
- $wp[[c_0]] T = \{\sigma \in \Sigma \mid \sigma[x \mapsto \sigma(x) - 2] \in T\}$
- Fixpoint iteration (with initial value $\bigsqcup \emptyset = \emptyset$):

$$\begin{aligned}\Psi(\emptyset) &= (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]] \emptyset) \cup S = \mathcal{B}[[x = 1]] \\ \Psi^2(\emptyset) &= (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]](\mathcal{B}[[x = 1]])) \cup S = \mathcal{B}[[x \in \{1, 3\}]] \\ \Psi^3(\emptyset) &= (\mathcal{B}[[x \neq 0 \wedge x \neq 1]] \cap wp[[c_0]](\mathcal{B}[[x \in \{1, 3\}]])) \cup S = \mathcal{B}[[x \in \{1, 3, 5\}]] \\ &\vdots\end{aligned}$$

$$\Rightarrow \text{fix}(\Psi) = \bigcup_{n \in \mathbb{N}} \Psi^n(\emptyset) = \{\sigma \in \Sigma \mid \sigma(x) \in \{1, 3, 5, \dots\}\}$$

Axiomatic Equivalence

Outline of Lecture 12

Recap: Completeness & Total Correctness

Soundness and Completeness of Hoare Logic for Total Correctness

Axiomatic Equivalence

Characteristic Assertions

Partial vs. Total Equivalence

Axiomatic Equivalence

Operational and Denotational Equivalence

Definition 4.1: $\mathcal{D}[\cdot]$: $Cmd \rightarrow (\Sigma \dashrightarrow \Sigma)$ given by

$$\mathcal{D}[c]\sigma = \sigma' \iff \langle c, \sigma \rangle \rightarrow \sigma'$$

Axiomatic Equivalence

Operational and Denotational Equivalence

Definition 4.1: $\mathcal{D}[\cdot]$: $Cmd \rightarrow (\Sigma \dashrightarrow \Sigma)$ given by

$$\mathcal{D}[c]\sigma = \sigma' \iff \langle c, \sigma \rangle \rightarrow \sigma'$$

Definition 4.2: Two statements $c_1, c_2 \in Cmd$ are **operationally equivalent** (notation: $c_1 \sim c_2$) if

$$\mathcal{D}[c_1] = \mathcal{D}[c_2]$$

Axiomatic Equivalence

Operational and Denotational Equivalence

Definition 4.1: $\mathcal{D}[\cdot]$: $Cmd \rightarrow (\Sigma \dashrightarrow \Sigma)$ given by

$$\mathcal{D}[c]\sigma = \sigma' \iff \langle c, \sigma \rangle \rightarrow \sigma'$$

Definition 4.2: Two statements $c_1, c_2 \in Cmd$ are **operationally equivalent** (notation: $c_1 \sim c_2$) if

$$\mathcal{D}[c_1] = \mathcal{D}[c_2]$$

Theorem 8.5: For every $c \in Cmd$,

$$\mathcal{D}[c] = \mathcal{C}[c]$$

(and thus operational and denotational equivalence coincide)

Axiomatic Equivalence

Axiomatic Equivalence I

In the axiomatic semantics, two statements have to be considered equivalent if they are **indistinguishable** w.r.t. (partial) correctness properties:

Definition 12.7 (Axiomatic equivalence)

Two statements $c_1, c_2 \in \text{Cmd}$ are called **axiomatically equivalent** (notation: $c_1 \approx c_2$) if, for all assertions $A, B \in \text{Assn}$,

$$\models \{A\} c_1 \{B\} \iff \models \{A\} c_2 \{B\}.$$

Axiomatic Equivalence

Axiomatic Equivalence I

In the axiomatic semantics, two statements have to be considered equivalent if they are **indistinguishable** w.r.t. (partial) correctness properties:

Definition 12.7 (Axiomatic equivalence)

Two statements $c_1, c_2 \in \text{Cmd}$ are called **axiomatically equivalent** (notation: $c_1 \approx c_2$) if, for all assertions $A, B \in \text{Assn}$,

$$\models \{A\} c_1 \{B\} \iff \models \{A\} c_2 \{B\}.$$

(later: total correctness yields same notion of equivalence)

Axiomatic Equivalence II

Example 12.8

We show that `while b do c end` \approx `if b then c; while b do c end else skip end`
(cf. Lemma 4.3).

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that `while b do c end` \approx `if b then c; while b do c end else skip end` (cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\}$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \end{aligned} \quad (\text{Theorem 10.4, 11.3})$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} && \text{(Theorem 10.4, 11.3)} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (cons))} \\ & \vdash \{C\} \text{ while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \end{aligned}$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} && \text{(Theorem 10.4, 11.3)} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (cons))} \\ & \vdash \{C\} \text{ while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (while))} \\ & \vdash \{C \wedge b\} c \{C\} \end{aligned}$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} && \text{(Theorem 10.4, 11.3)} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (cons))} \\ & \vdash \{C\} \text{ while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (while))} \\ & \vdash \{C \wedge b\} c \{C\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (seq), (skip))} \\ & \vdash \{C \wedge b\} c; \text{while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ & \vdash \{C \wedge \neg b\} \text{ skip } \{C \wedge \neg b\} \end{aligned}$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} && \text{(Theorem 10.4, 11.3)} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (cons))} \\ & \vdash \{C\} \text{ while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (while))} \\ & \vdash \{C \wedge b\} c \{C\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (seq), (skip))} \\ & \vdash \{C \wedge b\} c; \text{while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ & \vdash \{C \wedge \neg b\} \text{ skip } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (if))} \\ & \vdash \{C\} \text{ if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end } \{C \wedge \neg b\} \end{aligned}$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} && \text{(Theorem 10.4, 11.3)} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (cons))} \\ & \vdash \{C\} \text{ while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (while))} \\ & \vdash \{C \wedge b\} c \{C\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (seq), (skip))} \\ & \vdash \{C \wedge b\} c; \text{while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ & \vdash \{C \wedge \neg b\} \text{skip } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (if))} \\ & \vdash \{C\} \text{ if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end } \{C \wedge \neg b\} \\ \iff & \vdash \{A\} \text{ if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end } \{B\} && \text{(rule (cons))} \end{aligned}$$

Axiomatic Equivalence

Axiomatic Equivalence II

Example 12.8

We show that $\text{while } b \text{ do } c \text{ end} \approx \text{if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end}$
(cf. Lemma 4.3). Let $A, B \in \text{Assn}$:

$$\begin{aligned} & \models \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} \\ \iff & \vdash \{A\} \text{ while } b \text{ do } c \text{ end } \{B\} && \text{(Theorem 10.4, 11.3)} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (cons))} \\ & \vdash \{C\} \text{ while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (while))} \\ & \vdash \{C \wedge b\} c \{C\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (seq), (skip))} \\ & \vdash \{C \wedge b\} c; \text{while } b \text{ do } c \text{ end } \{C \wedge \neg b\} \\ & \vdash \{C \wedge \neg b\} \text{ skip } \{C \wedge \neg b\} \\ \iff & \text{ex. } C \in \text{Assn} \text{ such that } \models (A \Rightarrow C), \models (C \wedge \neg b \Rightarrow B), && \text{(rule (if))} \\ & \vdash \{C\} \text{ if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end } \{C \wedge \neg b\} \\ \iff & \vdash \{A\} \text{ if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end } \{B\} && \text{(rule (cons))} \\ \iff & \models \{A\} \text{ if } b \text{ then } c; \text{while } b \text{ do } c \text{ end else skip end } \{B\} && \text{(Theorem 10.4, 11.3)} \end{aligned}$$

Characteristic Assertions

Outline of Lecture 12

Recap: Completeness & Total Correctness

Soundness and Completeness of Hoare Logic for Total Correctness

Axiomatic Equivalence

Characteristic Assertions

Partial vs. Total Equivalence

Characteristic Assertions

Characteristic Assertions I

The following results are based of the following **encoding of states** by assertions:

Definition 12.9

Given a state $\sigma \in \Sigma$ and a finite subset of program variables $X \subseteq \text{Var}$, the **characteristic assertion of σ w.r.t. X** is given by

$$\text{state}(\sigma, X) := \bigwedge_{x \in X} (x = \underbrace{\sigma(x)}_{\in \mathbb{Z}}) \in \text{Assn}$$

(where $\text{state}(\sigma, \emptyset) := \text{true}$). Moreover, we let $\text{state}(\perp, X) := \text{false}$.

Characteristic Assertions

Characteristic Assertions I

The following results are based of the following **encoding of states** by assertions:

Definition 12.9

Given a state $\sigma \in \Sigma$ and a finite subset of program variables $X \subseteq \text{Var}$, the **characteristic assertion of σ w.r.t. X** is given by

$$\text{state}(\sigma, X) := \bigwedge_{x \in X} (x = \underbrace{\sigma(x)}_{\in \mathbb{Z}}) \in \text{Assn}$$

(where $\text{state}(\sigma, \emptyset) := \text{true}$). Moreover, we let $\text{state}(\perp, X) := \text{false}$.

Corollary 12.10

For all finite $X \subseteq \text{Var}$ and $\sigma \in \Sigma$,

$$\sigma \models \text{state}(\sigma, X)$$

Characteristic Assertions

Characteristic Assertions II

Programs and characteristic state assertions are obviously related as follows:

Corollary 12.11

Let $c \in \text{Cmd}$, and let $FV(c) \subseteq \text{Var}$ denote the set of all variables occurring in c .
Then, for every finite $X \supseteq FV(c)$ and $\sigma \in \Sigma$, $\models \{ \text{state}(\sigma, X) \} c \{ \text{state}(\mathcal{C}[[c]]\sigma, X) \}$.
If moreover $\mathcal{C}[[c]]\sigma \neq \perp$, then $\models \{ \text{state}(\sigma, X) \} c \{ \downarrow \text{state}(\mathcal{C}[[c]]\sigma, X) \}$.

Characteristic Assertions

Characteristic Assertions II

Programs and characteristic state assertions are obviously related as follows:

Corollary 12.11

Let $c \in \text{Cmd}$, and let $FV(c) \subseteq \text{Var}$ denote the set of all variables occurring in c . Then, for every finite $X \supseteq FV(c)$ and $\sigma \in \Sigma$, $\models \{state(\sigma, X)\} c \{state(\mathcal{C}[[c]]\sigma, X)\}$. If moreover $\mathcal{C}[[c]]\sigma \neq \perp$, then $\models \{state(\sigma, X)\} c \{\downarrow state(\mathcal{C}[[c]]\sigma, X)\}$.

Example 12.12 (Factorial program)

For $c := (y:=1; \text{ while } \neg(x=1) \text{ do } y:=y*x; x:=x-1 \text{ end})$,
 $X = \{x, y, z\} \supseteq FV(c) = \{x, y\}$, $\sigma(x) = 3$, $\sigma(y) = 0$, and $\sigma(z) = 1$, we obtain
 $state(\sigma, X) = (x=3 \wedge y=0 \wedge z=1)$ and $state(\mathcal{C}[[c]]\sigma, X) = (x=1 \wedge y=6 \wedge z=1)$
and thus $\models \{state(\sigma, X)\} c \{state(\mathcal{C}[[c]]\sigma, X)\}$.

Characteristic Assertions

Characteristic Assertions II

Programs and characteristic state assertions are obviously related as follows:

Corollary 12.11

Let $c \in \text{Cmd}$, and let $FV(c) \subseteq \text{Var}$ denote the set of all variables occurring in c . Then, for every finite $X \supseteq FV(c)$ and $\sigma \in \Sigma$, $\models \{state(\sigma, X)\} c \{state(\mathcal{C}[[c]]\sigma, X)\}$. If moreover $\mathcal{C}[[c]]\sigma \neq \perp$, then $\models \{state(\sigma, X)\} c \{\downarrow state(\mathcal{C}[[c]]\sigma, X)\}$.

Example 12.12 (Factorial program)

For $c := (y:=1; \text{while } \neg(x=1) \text{ do } y:=y*x; x:=x-1 \text{ end})$,
 $X = \{x, y, z\} \supseteq FV(c) = \{x, y\}$, $\sigma(x) = 3$, $\sigma(y) = 0$, and $\sigma(z) = 1$, we obtain
 $state(\sigma, X) = (x=3 \wedge y=0 \wedge z=1)$ and $state(\mathcal{C}[[c]]\sigma, X) = (x=1 \wedge y=6 \wedge z=1)$

and thus $\models \{state(\sigma, X)\} c \{state(\mathcal{C}[[c]]\sigma, X)\}$.

If $X \not\supseteq FV(c)$, then the claim does not hold: e.g., $\not\models \{y=0\} c \{y=6\}$!

Partial vs. Total Equivalence

Outline of Lecture 12

Recap: Completeness & Total Correctness

Soundness and Completeness of Hoare Logic for Total Correctness

Axiomatic Equivalence

Characteristic Assertions

Partial vs. Total Equivalence

Partial vs. Total Equivalence

Partial vs. Total Equivalence

Using characteristic state assertions, we can show that considering **total** rather than partial correctness properties yields the same notion of equivalence:

Theorem 12.13

Let $c_1, c_2 \in \text{Cmd}$. The following propositions are equivalent:

1. $\forall A, B \in \text{Assn} : \models \{A\} c_1 \{B\} \iff \models \{A\} c_2 \{B\}$
2. $\forall A, B \in \text{Assn} : \models \{A\} c_1 \{\Downarrow B\} \iff \models \{A\} c_2 \{\Downarrow B\}$

Partial vs. Total Equivalence

Partial vs. Total Equivalence

Using characteristic state assertions, we can show that considering **total** rather than partial correctness properties yields the same notion of equivalence:

Theorem 12.13

Let $c_1, c_2 \in \text{Cmd}$. The following propositions are equivalent:

1. $\forall A, B \in \text{Assn} : \models \{A\} c_1 \{B\} \iff \models \{A\} c_2 \{B\}$
2. $\forall A, B \in \text{Assn} : \models \{A\} c_1 \{\Downarrow B\} \iff \models \{A\} c_2 \{\Downarrow B\}$

Proof.

on the board □