Hoare Rules: Summary

\[ \vdash \{ A[x \mapsto e] \} \ x := e \ \{ A \} \]

\[ \vdash \{ A \} \ \text{if} \ b \ \text{then} \ c_1 \ \text{else} \ c_2 \ \{ B \} \]

\[ \vdash \{ A \wedge b \} \ c \ \{ A \} \]

\[ \vdash \{ A \} \ \text{while} \ b \ \text{do} \ c \ \{ A \wedge \neg b \} \]

\[ \vdash \{ A \} \ c_1 \ \{ C \} \quad \vdash \{ C \} \ c_2 \ \{ B \} \]

\[ \vdash A' \Rightarrow A \quad \vdash \{ A \} \ c \ \{ B \} \quad \vdash B \Rightarrow B' \]

\[ \vdash \{ A' \} \ c \ \{ B' \} \]
Hoare Rules: Conditional

\[ \vdash \{ A \land b \} \ c_1 \ \{ B \} \quad \vdash \{ A \land \neg b \} \ c_2 \ \{ B \} \]

\[ \vdash \{ A \} \text{ if } b \text{ then } c_1 \text{ else } c_2 \ \{ B \} \]

- Suppose we know \( A \) holds before if statement and want to show \( B \) holds afterwards
- At beginning of then branch, we know \( A \land b \), we prove \( B \) holds after executing the branch
- At beginning of else branch, we know \( A \land \neg b \), we prove \( B \) holds after executing the branch
Exercise

\[
\begin{align*}
\vdash \{ A[x \rightarrow e] \} \ x := e \ \{ A \} & \quad \vdash \{ A \land b \} \ c_1 \ \{ B \} \quad \vdash \{ A \land \neg b \} \ c_2 \ \{ B \} \\
\vdash \{ A \} \ if \ b \ then \ c_1 \ else \ c_2 \ \{ B \} \\
\vdash \{ A \} \ c_1 \ \{ C \} \quad \vdash \{ C \} \ c_2 \ \{ B \} & \quad \vdash A' \Rightarrow A \quad \vdash \{ A \} \ c \ \{ B \} \quad \vdash B \Rightarrow B' \\
\vdash \{ A \} \ c_1 \ ; \ c_2 \ \{ B \} & \quad \vdash \{ A' \} \ c \ \{ B' \}
\end{align*}
\]

• Under what condition \{ x > 0 \} holds after the following statement:

\[
\text{if } (x < 0) \text{ then } x := -x \text{ else } x := x
\]
Exercise

\[ \vdash \{ A \} \ x := e \ \{ A \} \quad \vdash \{ A \} \ \text{if } b \ \text{then } c_1 \ \text{else } c_2 \ \{ B \} \]

\[ \vdash \{ A \} \ \{ C \} \quad \vdash \{ C \} \ c_2 \ \{ B \} \quad \vdash A' \Rightarrow A \quad \vdash \{ A \} \ c \ \{ B \} \quad \vdash B \Rightarrow B' \]

\[ \vdash \{ A \} \ c_1 ; \ c_2 \ \{ B \} \]

- Under what condition \( \{ x > 0 \} \) holds after the following statement:

\[
\text{if } (x < 0) \ \text{then } x := -x \ \text{else } x := x
\]

**Solution:** \( x \) should not be 0 initially

\[ \vdash \{ (x < 0) \} \ x := -x \ \{ x > 0 \} \]

\[ \vdash \{ (x \neq 0) \land (x < 0) \} \ x := -x \ \{ x > 0 \} \]

\[ \vdash \{ (x > 0) \} \ x := -x \ \{ x > 0 \} \]

\[ \vdash \{ (x \neq 0) \land (x \geq 0) \} \ x := x \ \{ x > 0 \} \]

\[ \vdash \{ x \neq 0 \} \ \text{if } (x < 0) \ \text{then } x := -x \ \text{else } x := x + 1 \ \{ x > 0 \} \]
To prove a sequence \( \{A\} \; c_1; c_2 \; \{B\} \) we must find an intermediate assertion \( C \)

- Implied by \( A \) after \( c_1 \) and implying \( B \) after \( c_2 \)
  - (often denoted \( \{A\} \; c_1 \; \{C\} \; c_2 \; \{B\} \) )
Exercise

⊢ \{ A \} \ c_1 \ \{ C \} \quad \vdash \{ C \} \ c_2 \ \{ B \} \\
\vdash \{ A \} \ c_1 ; \ c_2 \ \{ B \}

• What is the intermediate assertion to prove the following Hoare triple?

\{ \text{true} \} \ x := 1; \ y := x \ \{ x = 1 \land y = 1 \}
What is the intermediate assertion to prove the following Hoare triple?

\[
\{ \text{true} \} \ x := 1; \ y := x \ \{ x = 1 \land y = 1 \}
\]

**Solution:**  \( (x = 1) \)
Hoare Rules: Consequence

Pre-condition strengthening, Post-condition weakening

\[ \vdash A' \Rightarrow A \] \[ \vdash \{A\} \ c \ \{B\} \vdash B \Rightarrow B' \]

\[ \vdash \{A'\} \ c \ \{B'\} \]

- Suppose we can prove \( \{x \geq 0 \land y < 2\} \ c \ \{x = 0 \land y \leq 0\} \)
- Which of the following Hoare triples can we prove?

\[
\begin{align*}
\{x \geq 0 \land y \leq 0\} & \ c \ \{x = 0 \land y \leq 0\} \\
\{x \geq 0 \land y \geq 0\} & \ c \ \{x = 0 \land y \leq 0\} \\
\{x = 5\} & \ c \ \{y \leq 1\}
\end{align*}
\]
Pre-condition strengthening, Post-condition weakening

\[ \vdash A' \Rightarrow A \vdash \{A\} \ c \ \{B\} \vdash B \Rightarrow B' \]
\[ \vdash \{A'\} \ c \ \{B'\} \]

- Suppose we can prove \( \{x \geq 0 \land y < 2\} \ c \ \{x = 0 \land y \leq 0\} \)
- Which of the following Hoare triples can we prove?

\[
\begin{align*}
\{x \geq 0 \land y \leq 0\} & \quad c \quad \{x = 0 \land y \leq 0\} \quad \checkmark \\
\{x \geq 0 \land y \geq 0\} & \quad c \quad \{x = 0 \land y \leq 0\} \\
\{x = 5\} & \quad c \quad \{y \leq 1\}
\end{align*}
\]
Hoare Rules: Consequence

Pre-condition strengthening, Post-condition weakening

\[ \vdash A' \Rightarrow A \vdash \{A\} \ c \ \{B\} \ \vdash \ B \Rightarrow B' \]
\[ \vdash \{A'\} \ c \ \{B'\} \]

- Suppose we can prove \( \{x \geq 0 \land y < 2\} \ c \ \{x = 0 \land y \leq 0\} \)
- Which of the following Hoare triples can we prove?

<table>
<thead>
<tr>
<th>pre-condition</th>
<th>post-condition</th>
<th>result</th>
</tr>
</thead>
<tbody>
<tr>
<td>{x \geq 0 \land y \leq 0}</td>
<td>c \ {x = 0 \land y \leq 0}</td>
<td>✓</td>
</tr>
<tr>
<td>{x \geq 0 \land y \geq 0}</td>
<td>c \ {x = 0 \land y \leq 0}</td>
<td>✗</td>
</tr>
<tr>
<td>{x = 5}</td>
<td>c \ {y \leq 1}</td>
<td></td>
</tr>
</tbody>
</table>
Suppose we can prove \( \{x \geq 0 \land y < 2\} \ c \ \{x = 0 \land y \leq 0\} \)

Which of the following Hoare triples can we prove?

- \( \{x \geq 0 \land y \leq 0\} \ c \ \{x = 0 \land y \leq 0\} \) \quad \checkmark
- \( \{x = 0 \land y \leq 0\} \ c \ \{x = 0 \land y \leq 0\} \) \quad \times
- \( \{x = 5\} \ c \ \{y \leq 1\} \) \quad \times
Hoare Rules: Loops

\[ \vdash \{ A \wedge b \} \ c \ \{ A \} \]
\[ \vdash \{ A \} \ \text{while} \ b \ \text{do} \ c \ \{ A \wedge \neg b \} \]

- Assertion \( A \) is a loop invariant: assertion that remains true before and after every iteration of the loop.

\[ \vdash \{ A \wedge b \} \ c \ \{ A \} \]

- Both a pre-condition for the loop (holds before the first iteration) and a post-condition for the loop (holds after the last iteration)
Loop Invariant:

- What has been done so far and what remains to be done
- That nothing has been done initially
- That nothing remains to be done when $b$ is false
Example

- Consider the statement \((x, n \in \mathbb{Z})\)

\[ S = \text{while } x < n \text{ do } x := x + 1 \]

- Prove validity of \(\{x \leq n\} S \{x \geq n\}\)
- First Step: What is appropriate loop invariant?
Example

- Consider the statement \((x, n \in \mathbb{Z})\)

\[
S = \text{while } x < n \text{ do } x := x + 1
\]

- Prove validity of \(\{x \leq n\} \ S \ \{x \geq n\}\)

- First Step: What is appropriate loop invariant? \(x \leq n\)

- First, we need to prove \(\{x \leq n \land x < n\} \ x := x + 1 \ \{x \leq n\}\)

- Required proof rules: assignment, precondition strengthening

\[
\begin{align*}
\vdash \{x \leq n[x \mapsto x + 1]\} & \quad x := x + 1 \ \{x \leq n\} \\
\vdash \{x + 1 \leq n\} & \quad x := x + 1 \ \{x \leq n\} \\
\vdash \{x \leq n \land x < n\} & \quad x := x + 1 \ \{x \leq n\}
\end{align*}
\]

\[
x \leq n \land x < n \Rightarrow x + 1 \leq n
\]
• Let’s instantiate proof rule for `while` with this loop invariant:

\[
\begin{align*}
\vdash \{ x \leq n \land x < n \} & \quad \begin{array}{c}
x := x + 1 \{ x \leq n \} \\
\end{array} \\
\vdash \{ x \leq n \} & \quad \text{while } x < n \text{ do } x := x + 1 \{ x \leq n \land \neg (x < n) \}
\end{align*}
\]

• Recall: We wanted to prove the Hoare triple

\[
\{ x \leq n \} \ S \ { x \geq n}\]

• In addition to proof rule for `while`, what other rule do we need?
• Let’s instantiate proof rule for \texttt{while} with this loop invariant:

\[
\frac{\vdash \{x \leq n \land x < n\} \ x := x + 1 \ \{x \leq n\}}{\vdash \{x \leq n\} \ while \ x < n \ do \ x := x + 1 \ \{x \leq n \land \neg(x < n)\}}
\]

• Recall: We wanted to prove the Hoare triple

\[
\{x \leq n\} \ S \ \{x \geq n\}
\]

• In addition to proof rule for \texttt{while}, what other rule do we need? postcondition weakening
To prove the Hoare triple \{A\} while \(b\) do \(c\) \{B\}

- Find \(I\) and prove it is an invariant: \(\vdash \{b \land I\} \ c \ \{I\}\)
- Prove \(I\) is \textit{true} at the start: \(A \Rightarrow I\)
- Prove \(B\) is \textit{true} after the loop: \(I \land \neg b \Rightarrow B\)
Exercise

• Suppose we add a for loop construct to IMP

\[
\text{for } x := e_1 \text{ until } e_2 \text{ do } S
\]

• Initializes \( x \) to \( e_1 \), increments \( x \) by 1 in each iteration and terminates when \( x > e_2 \)

• Write a proof rule for this for loop construct