Programming Languages: Imperative Program Construction
Practicals 0: Non-Looping Constructs and Weakest Precondition

Shin-Cheng Mu

Autumn Term, 2022

Guarded Command Language Basics

1. Which of the following Hoare triples hold?
   (a) $\{x = 7\} skip \{odd\ x\}$;
   (b) $\{x > 60\} x := x \times 2 \{x > 100\}$;
   (c) $\{x > 40\} x := x \times 2 \{x > 100\}$;
   (d) $\{true\}$**if** $x \leqslant y \rightarrow y := y - x \mid x \geqslant y \rightarrow x := x - y$ **fi**$\{x \geqslant 0 \wedge y \geqslant 0\}$;
   (e) $\{even\ x \wedge even\ y\}$**if** $x \leqslant y \rightarrow y := y - x \mid x \geqslant y \rightarrow x := x - y$ **fi**$\{even\ x \wedge even\ y\}$.

Solution: As the first exercise I expect merely that you answer by informal reasoning. What follows is the more formal approach which you will learn later.

(a) The Hoare triple holds because:

$$
\begin{aligned}
& wp\ skip\ (odd\ x) \\
\equiv\ & \quad \{ \text{ definition of } wp \ \} \\
& odd\ x \\
\Leftarrow\ & x = 7 \ .
\end{aligned}
$$

(b) The Hoare triple holds because:

$$
\begin{aligned}
& wp\ (x := x \times 2)\ (x > 100) \\
\equiv\ & \quad \{ \text{ definition of } wp \ \} \\
& x \times 2 > 100 \\
\Leftarrow\ & x > 60 \ .
\end{aligned}
$$

(c) The Hoare triple does not hold because:

$$
\begin{aligned}
& wp\ (x := x \times 2)\ (x > 100) \\
\equiv\ & x \times 2 > 100 \\
\not\Leftarrow\ & x > 40 \ .
\end{aligned}
$$

(d) The annotated **if** statement is

$$
\begin{array}{l}
\{True\} \\
\textbf{if}\ x \leqslant y \rightarrow \{x \leqslant y\}\ y := y - x\ \{x \geqslant 0 \wedge y \geqslant 0\} \\
\quad x \geqslant y \rightarrow \{x \geqslant y\}\ x := x - y\ \{x \geqslant 0 \wedge y \geqslant 0\} \\
\textbf{fi} \\
\{x \geqslant 0 \wedge y \geqslant 0\} \ .
\end{array}
$$

That $x \leqslant y \vee x \geqslant y$ certainly holds. For the Hoare triple in the first branch we reason:

$$
\begin{aligned}
& (x \geqslant 0 \wedge y \geqslant 0)[y \backslash y - x] \\
\equiv\ & x \geqslant 0 \wedge y - x \geqslant 0 \\
\equiv\ & x \geqslant 0 \wedge x \leqslant y \\
\not\Leftarrow\ & x \leqslant y \ .
\end{aligned}
$$

The situation with the other branch is similar. The bottom line is that the initial Hoare triple does *not* hold.

The initial Hoare triple would be true if the precondition were $x \geqslant 0 \wedge y \geqslant 0$.

(e) The annotated **if** statement is

$$
\begin{aligned}
& \{\mathit{even}\ x \wedge \mathit{even}\ y\} \\
& \mathbf{if}\ x \leqslant y \rightarrow \{\mathit{even}\ x \wedge \mathit{even}\ y \wedge x \leqslant y\}\ y := y - x\ \{\mathit{even}\ x \wedge \mathit{even}\ y\} \\
& \quad\ x \geqslant y \rightarrow \{\mathit{even}\ x \wedge \mathit{even}\ y \wedge x \geqslant y\}\ x := x - y\ \{\mathit{even}\ x \wedge \mathit{even}\ y\} \\
& \mathbf{fi} \\
& \{\mathit{even}\ x \wedge \mathit{even}\ y\} \ .
\end{aligned}
$$

That $x \leqslant y \vee x \geqslant y$ certainly holds. For the Hoare triple in the first branch we reason:

$$
\begin{aligned}
& (\mathit{even}\ x \wedge \mathit{even}\ y)[y \backslash y - x] \\
\equiv\ & \mathit{even}\ x \wedge \mathit{even}\ (y - x) \\
\equiv\ & \mathit{even}\ x \wedge \mathit{even}\ y \\
\Leftarrow\ & \mathit{even}\ x \wedge \mathit{even}\ y \wedge x \leqslant y \ .
\end{aligned}
$$

The situation with the other branch is similar. The bottom line is that the initial Hoare triple does hold.

2. Is it always true that $\{\mathit{True}\}\ x := E\ \{x = E\}$? If you think the answer is yes, explain why. If your answer is no, give a counter example.

Solution: No. For a counterexample, let E be $x + 1$.

When do we do have the property that $\{\mathit{True}\}\ x := E\ \{x = E\}$? Since $(x = E)[x \backslash E] \equiv (E = E\ [x \backslash E])$, the Hoare triple holds if and only if $E = E\ [x \backslash E]$. Examples of such E include those that do not contain x, or those that are idempotent funtions on x, for example $E = 0 \uparrow x$.

The actual forward rule for assignment (due to Floyd) is:

$$\{P\}\ x := E\ \{(\exists\ x_0 :: x = E\ [x \backslash x_0] \wedge P\ [x \backslash x_0])\} \ ,$$

where x_0 is a fresh name.

3. Verify:

$$
\begin{aligned}
& \{x = X \wedge y = Y\} \\
& x := x \not\Leftrightarrow y \\
& y := x \not\Leftrightarrow y \\
& x := x \not\Leftrightarrow y \\
& \{x = Y \wedge y = X\}
\end{aligned}
$$

where x and y are boolean and $(\not\Leftrightarrow)$ is the “not equal” or “exclusive or” operator. In fact, the code above works for any $(\otimes)$ that satisfies the properties that for all a, b, and c:

$$\begin{array}{ll} \text{associative}: & a \otimes (b \otimes c) = (a \otimes b) \otimes c \,, \\ \text{unipotent}: & a \otimes a = 1 \,, \end{array}$$

where 1 is the unit of $(\otimes)$, that is, $1 \otimes b = b = b \otimes 1$.

Solution: The annotated program is:

$$\begin{array}{l} \{x = X \wedge y = Y, \mathrm{Pf}_2\} \\ x := x \otimes y \\ \{y = Y \wedge x \otimes y = X, \mathrm{Pf}_1\} \\ y := x \otimes y \\ \{x \otimes y = Y \wedge y = X\} \\ x := x \otimes y \\ \{x = Y \wedge y = X\} \ . \end{array}$$

Pf_1:

$$\begin{array}{ll} & (x \otimes y = Y \wedge y = X)\,[x \otimes y \,/\, y] \\ \equiv & x \otimes (x \otimes y) = Y \wedge x \otimes y = X \\ \equiv & \quad \{ (\otimes) \text{ associative} \} \\ & (x \otimes x) \otimes y = Y \wedge x \otimes y = X \\ \equiv & \quad \{ \text{unipotence} \} \\ & 1 \otimes y = Y \wedge x \otimes y = X \\ \equiv & \quad \{ \text{identity} \} \\ & y = Y \wedge x \otimes y = X \ . \end{array}$$

Pf_2:

$$\begin{array}{ll} & (y = Y \wedge x \otimes y = X)\,[x \otimes y \,/\, x] \\ \equiv & y = Y \wedge (x \otimes y) \otimes y = X \\ \equiv & \quad \{ (\otimes) \text{ associative} \} \\ & y = Y \wedge x \otimes (y \otimes y) = X \\ \equiv & \quad \{ \text{unipotence} \} \\ & y = Y \wedge x \otimes 1 = X \\ \equiv & \quad \{ \text{identity} \} \\ & y = Y \wedge x = X \ . \end{array}$$

4. Verify the following program:

$$\begin{array}{l} \mathbf{var}\ r, b : \mathit{Int} \\ \{0 \leqslant r < 2 \times b\} \\ \mathbf{if}\ b \leqslant r \rightarrow r := r - b \\ \mid\ r < b \rightarrow \mathit{skip} \\ \mathbf{fi} \\ \{0 \leqslant r < b\} \end{array}$$

Solution: The annotated program is:

$$
\begin{array}{l}
\textbf{var } r, b : \mathit{Int} \\
\{0 \leqslant r < 2 \times b\} \\
\textbf{if } b \leqslant r \rightarrow \{0 \leqslant r < 2 \times b \wedge b \leqslant r\}\ r := r - b\ \{0 \leqslant r < b, \mathrm{Pf}_1\} \\
\ \mid\ r < b \rightarrow \{0 \leqslant r < 2 \times b \wedge r < b\}\ \mathit{skip}\ \{0 \leqslant r < b, \mathrm{Pf}_2\} \\
\textbf{fi} \\
\{0 \leqslant r < b, \mathrm{Pf}_3\}
\end{array}
$$

Pf_1. We reason:

$$
\begin{array}{ll}
 & (0 \leqslant r < b)\ [r \backslash r - b] \\
\equiv & 0 \leqslant r - b < b \\
\equiv & b \leqslant r < 2 \times b \\
\Leftarrow & 0 \leqslant r < 2 \times b \wedge b \leqslant r\ \ .
\end{array}
$$

Pf_2. Trivial.

Pf_3. Certainly any proposition implies $b \leqslant r \vee r < b$.

5. Verify:

$$
\begin{array}{l}
\textbf{var } x, y : \mathit{Int} \\
\{\mathit{True}\} \\
x, y := x \times x, y \times y \\
\textbf{if } x \geqslant y \rightarrow x := x - y \\
\ \mid\ y \geqslant x \rightarrow y := y - x \\
\textbf{fi} \\
\{x \geqslant 0 \wedge y \geqslant 0\}\ \ .
\end{array}
$$

Solution: For brevity we abbreviate $x \geqslant 0\ \wedge\ y \geqslant 0$ to P. The fully annotated program could be:

$$
\begin{array}{l}
\{\mathit{True}\} \\
x, y := x \times x, y \times y \\
\{P, \mathrm{Pf}_4\} \\
\textbf{if } x \geqslant y \rightarrow \{x \geqslant y \wedge P\}\ x := x - y\ \{P, \mathrm{Pf}_1\} \\
\ \mid\ y \geqslant x \rightarrow \{y \geqslant x \wedge P\}\ y := y - x\ \{P, \mathrm{Pf}_2\} \\
\textbf{fi} \\
\{P, \mathrm{Pf}_3\}\ \ .
\end{array}
$$

To verify the **if** branching, we check that

Pf_1. $\{x \geqslant y\ \wedge\ P\}\ x := x - y\ \{P\}$. The Hoare triple is valid because

$$
\begin{array}{ll}
 & (x \geqslant 0\ \wedge\ y \geqslant 0)[x \backslash x - y] \\
\Leftrightarrow & x - y \geqslant 0\ \wedge\ y \geqslant 0 \\
\Leftrightarrow & x \geqslant y \wedge\ y \geqslant 0 \\
\Leftarrow & x \geqslant y\ \wedge\ x \geqslant 0\ \wedge\ y \geqslant 0.
\end{array}
$$

Pf$_2$. $\{y \geqslant x \ \wedge \ P\}\, y := y - x\, \{P\}$. Omitted.

Pf$_3$. And indeed $x \geqslant y \ \vee \ y \geqslant x$ always holds, thus $P \Rightarrow x \geqslant y \ \vee \ y \geqslant x$.

Do not forget that we have yet to verify $\{true\}\, x, y := x \times x, y \times y\, \{P\}$, which is not difficult either:

Pf$_4$.

$$\begin{aligned}
& (x \geqslant 0 \ \wedge \ y \geqslant 0)[x, y \backslash x \times x, y \times y] \\
\Leftrightarrow \ & x \times x \geqslant 0 \ \wedge \ y \times y \geqslant 0 \\
\Leftrightarrow \ & true.
\end{aligned}$$

6. Verify:

$$\begin{array}{l}
\textbf{var } a, b : Bool \\
\{True\} \\
\textbf{if } \neg\, a \vee b \rightarrow a := \neg\, a \\
\ \mid \ a \vee \neg\, b \rightarrow b := \neg\, b \\
\textbf{fi} \\
\{a \vee b\} \ \ .
\end{array}$$

Solution:

$$\begin{array}{l}
\textbf{var } a, b : Bool \\
\{True\} \\
\textbf{if } \neg\, a \vee b \rightarrow \{\neg\, a \vee b\}\, a := \neg\, a\, \{a \vee b, \mathrm{Pf}_1\} \\
\ \mid \ a \vee \neg\, b \rightarrow \{a \vee \neg\, b\}\, b := \neg\, b\, \{a \vee b, \mathrm{Pf}_2\} \\
\textbf{fi} \\
\{a \vee b, \mathrm{Pf}_3\} \ \ .
\end{array}$$

Pf$_1$. To verify the first branch:

$$\begin{aligned}
& (a \vee b)[a \backslash \neg a] \\
\equiv \ & \neg a \vee b.
\end{aligned}$$

Pf$_2$. The other branch is similar.

Pf$_3$. Certainly $true \Rightarrow \neg a \vee b \vee a \vee \neg b$.

7. Assuming that x, y, and z are integers, prove the following
   (a) $\{True\}\, \textbf{if } x \geqslant 1 \rightarrow x := x + 1 \mid x \leqslant 1 \rightarrow x := x - 1 \textbf{ fi}\, \{x \neq 1\}$.
   (b) $\{True\}\, \textbf{if } x \geqslant y \rightarrow skip \mid y \geqslant x \rightarrow x, y := y, x \textbf{ fi}\, \{x \geqslant y\}$.
   (c) $\{x = 0\}\, \textbf{if } True \rightarrow x := 1 \mid True \rightarrow x := -1\, \{x = 1 \vee x = -1\}$.
   (d) $\{A = x \times y + z\}\, \textbf{if } even\ x \rightarrow x, y := x \,/\, 2, y \times 2 \mid True \rightarrow y, z := y - 1, z + x\, \{A = x \times y + z\}$.

Solution: The annotated program is

$$\begin{array}{l}
\{A = x \times y + z\} \\
\textbf{if } \mathit{even}\ x \rightarrow \{A = x \times y + z \wedge \mathit{even}\ x\}\ x, y := x \,/\, 2, y \times 2\ \{A = x \times y + z, \mathrm{Pf}_0\} \\
\ \mid\ \mathit{True} \quad \rightarrow \{A = x \times y + z\}\ y, z := y - 1, z + x\ \{A = x \times y + z, \mathrm{Pf}_1\} \\
\textbf{fi} \\
\{A = x \times y + z, \mathrm{Pf}_2\}
\end{array}$$

Pf_0: We reason:

$$\begin{array}{ll}
 & (A = x \times y + z)[x, y \backslash x \,/\, 2, y \times 2] \\
\equiv & A = (x \,/\, 2) \times (y \times 2) + z \\
\Leftarrow & A = x \times y + z \wedge \mathit{even}\ x \ \ .
\end{array}$$

Pf_2: We reason:

$$\begin{array}{ll}
 & (A = x \times y + z)[y, z \backslash y - 1, z + x] \\
\equiv & A = x \times (y - 1) + (z + x) \\
\Leftarrow & A = x \times y + z \ \ .
\end{array}$$

Pf_2: Certainly $P \Rightarrow Q \wedge \mathit{True}$ for any P and Q.

(e) $\{x \times y = 0 \wedge y \leqslant x\}$ **if** $y < 0 \rightarrow y := -y \mid y = 0 \rightarrow x := -1\ \{x < y\}$.

Solution: The annotated program is

$$\begin{array}{l}
\{x \times y = 0 \wedge y \leqslant x\} \\
\textbf{if } y < 0 \rightarrow \{x \times y = 0 \wedge y \leqslant x \wedge y < 0\}\ y := -y\ \{x < y, \mathrm{Pf}_0\} \\
\ \mid\ y = 0 \ \rightarrow \{x \times y = 0 \wedge y \leqslant x \wedge y = 0\}\ x := -1\ \{x < y, \mathrm{Pf}_1\} \\
\textbf{fi} \\
\{x < y, \mathrm{Pf}_2\}
\end{array}$$

Pf_0: Note that $x \times y = 0$ equivals $x = 0 \vee y = 0$. Therefore

$$\begin{array}{ll}
 & x \times y = 0 \wedge y \leqslant x \wedge y < 0 \\
\equiv & (x = 0 \vee y = 0) \wedge y \leqslant x \wedge y < 0 \\
\equiv & \quad \{ \text{ distributivity } \} \\
 & (x = 0 \wedge y \leqslant x \wedge y < 0) \vee (y = 0 \wedge y \leqslant x \wedge y < 0) \\
\equiv & \quad \{ \text{ since } (y = 0 \wedge y \leqslant x \wedge y < 0) \equiv \mathit{False} \ \} \\
 & x = 0 \wedge y \leqslant x \wedge y < 0 \\
\equiv & x = 0 \wedge y < 0 \ \ .
\end{array}$$

To prove the Hoare triple we reason:

$$\begin{array}{ll}
 & (x < y)[y \backslash -y] \\
\equiv & x < -y \\
\Leftarrow & x = 0 \wedge y < 0 \ \ .
\end{array}$$

Pf_1: We reason:

$$\begin{array}{ll}
 & (x < y)[x \backslash -1] \\
\equiv & -1 < y \\
\Leftarrow & x \times y = 0 \wedge y \leqslant x \wedge y = 0 \ \ .
\end{array}$$

Pf$_2$: We reason:

$$
\begin{aligned}
& x \times y = 0 \wedge y \leqslant x \\
\equiv\ & (x = 0 \vee y = 0) \wedge y \leqslant x \\
\equiv\ & \quad \{ \text{ distributivity } \} \\
& (x = 0 \wedge y \leqslant x) \vee (y = 0 \wedge y \leqslant x) \\
\Rightarrow\ & y < 0 \vee y = 0 \quad .
\end{aligned}
$$

Weakest Precondition of Simple Statements

8. Given below is a list of statements and predicates. What are the weakest precondition for the predicates to be true after the statement?
   (a) $x := x \times 2$, $x > 100$;
   (b) $x := x \times 2$, *even* x;
   (c) $x := x \times 2$, $x > 100 \wedge \mathit{even}\ x$;
   (d) $x := x \times 2$, *odd* x.
   (e) *skip*, *odd* x.

Solution:

(a) $x \times 2 > 100$, that is, $x > 50$.

(b) *even* $(x \times 2)$, which simplifies to *True*.

(c) $x \times 2 > 100 \wedge \mathit{even}\ (x \times 2)$, that is, $x > 50$.

(d) *odd* $(x \times 2)$, that is, *False*.

(e) *odd* x.

9. Determine the weakest P that satisfies
   (a) $\{P\}\ x := x + 1; x := x + 1\ \{x \geqslant 0\}$.
   (b) $\{P\}\ x := x + y; y := 2 \times x\ \{y \geqslant 0\}$.
   (c) $\{P\}\ x := y; y := x\ \{x = A \wedge y = B\}$.
   (d) $\{P\}\ x := E; x := E\ \{x = E\}$.

Solution:

(a)

$$
\begin{aligned}
& \mathit{wp}\ (x := x + 1; x := x + 1)\ (x \geqslant 0) \\
=\ & \mathit{wp}\ (x := x + 1)\ (\mathit{wp}\ (x := x + 1)\ (x \geqslant 0)) \\
=\ & \mathit{wp}\ (x := x + 1)\ (x + 1 \geqslant 0) \\
=\ & (x + 1) + 1 \geqslant 0 \\
=\ & x \geqslant -2 \quad .
\end{aligned}
$$

(b)

$$\begin{aligned}
& wp\ (x := x + y; y := 2 \times x)\ (y \geqslant 0) \\
= {} & wp\ (x := x + y)\ (wp\ (y := 2 \times x)\ (y \geqslant 0)) \\
= {} & wp\ (x := x + y)\ (2 \times x \geqslant 0) \\
= {} & 2 \times (x + y) \geqslant 0 \ .
\end{aligned}$$

(c)

$$\begin{aligned}
& wp\ (x := y; y := x)\ (x = A \wedge y = B) \\
\equiv {} & wp\ (x := y)\ (wp\ (y := x)\ (x = A \wedge y = B)) \\
\equiv {} & wp\ (x := y)\ (x = A \wedge x = B) \\
\equiv {} & y = A \wedge y = B \\
\equiv {} & y = A = B \ .
\end{aligned}$$

(d)

$$\begin{aligned}
& wp\ (x := E; x := E)\ (x = E) \\
\equiv {} & wp\ (x := E)\ (wp\ (x := E)\ (x = E)) \\
\equiv {} & wp\ (x := E)\ ((x = E)[x \backslash E]) \\
\equiv {} & wp\ (x := E)\ (E = E[x \backslash E]) \\
\equiv {} & (E = E[x \backslash E])[x \backslash E] \\
\equiv {} & E[x \backslash E] = (E[x \backslash E])[x \backslash E] \ .
\end{aligned}$$

The equation certainly does not hold in general. One example where it does hold is $E = (-x) \uparrow 0$, for which we have:

$$\begin{aligned}
& E[x \backslash E] \\
= {} & (-((-x) \uparrow 0)) \uparrow 0 \\
= {} & (x \downarrow 0) \uparrow 0 \\
= {} & 0 \\
= {} & (-0) \uparrow 0 \\
= {} & (-((-((-x) \uparrow 0)) \uparrow 0)) \uparrow 0 \\
= {} & (E[x \backslash E])[x \backslash E] \ .
\end{aligned}$$

Let me know if you have a more interesting E.

10. What is the weakest P such that the following holds?

var $x : Int$
$\{P\}$
$x := x + 1$
if $x > 0 \rightarrow x := x + 1$
$\mid\ x < 0 \rightarrow x := x + 2$
$\mid\ x = 1 \rightarrow skip$
fi
$\{x \geqslant 1\}$.

Solution: Denote the **if** statement by IF. The aim is to compute $wp\ (x := x + 1; \mathrm{IF})\ (x \geqslant 1)$.

Recall the definition of wp for **if**. We have

$$\begin{aligned}
wp\ \mathrm{IF}\ (x \geqslant 1) = {} & (x > 0 \Rightarrow wp\ (x := x + 1)\ (x \geqslant 1)) \wedge \\
& (x < 0 \Rightarrow wp\ (x := x + 2)\ (x \geqslant 1)) \wedge \\
& (x = 1 \Rightarrow wp\ skip\ (x \geqslant 1)) \wedge \\
& (x > 0 \vee x < 0 \vee x = 1) \ .
\end{aligned}$$

We calculate the four conjuncts separately:

- $$\begin{aligned} & x > 0 \Rightarrow wp\ (x := x + 1)\ (x \geqslant 1) \\ \equiv\ & x > 0 \Rightarrow x + 1 \geqslant 1 \\ \equiv\ & x > 0 \Rightarrow x \geqslant 0 \\ \equiv\ & \mathit{True}\ . \end{aligned}$$

- $$\begin{aligned} & x < 0 \Rightarrow wp\ (x := x + 2)\ (x \geqslant 1) \\ \equiv\ & x < 0 \Rightarrow x + 2 \geqslant 1 \\ \equiv\ & x < 0 \Rightarrow x \geqslant -1 \\ \equiv\ & \quad \{\ (P \Rightarrow Q) = (\neg\ P \vee Q)\ \} \\ & x \geqslant 0 \vee x \geqslant -1 \\ \equiv\ & x \geqslant -1\ \ . \end{aligned}$$

- $$\begin{aligned} & x = 1 \Rightarrow wp\ \mathit{skip}\ (x \geqslant 1) \\ \equiv\ & x = 1 \Rightarrow x \geqslant 1 \\ \equiv\ & \mathit{True}\ \ . \end{aligned}$$

- Furthermore, $x > 0 \vee x < 0 \vee x = 1$ simplifies to $x \neq 0$.

Therefore,

$$\begin{aligned} & wp\ \mathrm{IF}\ (x \geqslant 1) \\ =\ & \mathit{True} \wedge x \geqslant -1 \wedge \mathit{True} \wedge x \neq 0 \\ =\ & x \geqslant -1 \wedge x \neq 0\ \ . \end{aligned}$$

Finally, recall what we want to compute:

$$\begin{aligned} & wp\ (x := x + 1; \mathrm{IF})\ (x \geqslant 1) \\ =\ & wp\ (x := x + 1)\ (wp\ \mathrm{IF}\ (x \geqslant 1)) \\ =\ & wp\ (x := x + 1)\ (x \geqslant -1 \wedge x \neq 0) \\ =\ & x + 1 \geqslant -1 \wedge x + 1 \neq 0) \\ =\ & x \geqslant -2 \wedge x \neq -1\ \ . \end{aligned}$$

11. Two programs S_0 and S_1 are equivalent if, for all Q, $wp\ S_0\ Q = wp\ S_1\ Q$. Show that the two following programs are equivalent.

$$\begin{aligned} & \textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1\ \textbf{fi}; S \\ & \textbf{if}\ B_0 \rightarrow S_0; S \mid B_1 \rightarrow S_1; S\ \textbf{fi} \end{aligned}$$

Solution:

$$
\begin{aligned}
& wp\ (\textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1\ \textbf{fi}; S)\ Q \\
= \quad & \{ \text{ definition of } wp \ \} \\
& wp\ (\textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1\ \textbf{fi})\ (wp\ S\ Q) \\
= \quad & \{ \text{ definition of } wp \ \} \\
& (B_0 \Rightarrow wp\ S_0\ (wp\ S\ Q)) \wedge \\
& (B_1 \Rightarrow wp\ S_1\ (wp\ S\ Q)) \wedge (B_0 \vee B_1) \\
= \quad & \{ \text{ definition of } wp \ \} \\
& (B_0 \Rightarrow wp\ (S_0; S)\ Q) \wedge \\
& (B_1 \Rightarrow wp\ (S_1; S)\ Q) \wedge (B_0 \vee B_1) \\
= \quad & \{ \text{ definition of } wp \ \} \\
& wp\ (\textbf{if}\ B_0 \rightarrow S_0; S \mid B_1 \rightarrow S_1; S\ \textbf{fi})\ Q \ \ .
\end{aligned}
$$

12. Consider the two programs:

$$
\begin{aligned}
\mathsf{IF}_0 &= \textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1\ \textbf{fi} \ \ , \\
\mathsf{IF}_1 &= \textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \wedge \neg B_0 \rightarrow S_1\ \textbf{fi} \ \ .
\end{aligned}
$$

Show that for all Q, $wp\ \mathsf{IF}_0\ Q \Rightarrow wp\ \mathsf{IF}_1\ Q$.

Solution: Firstly, we show that $B_0 \vee (B_1 \wedge \neg B_0) = B_0 \vee B_1$.

$$
\begin{aligned}
& B_0 \vee (B_1 \wedge \neg B_0) \\
= \quad & \{ \text{ distributivity } \} \\
& (B_0 \vee B_1) \wedge (B_0 \vee \neg B_0) \\
= \ & (B_0 \vee B_1) \wedge \mathit{True} \\
= \ & B_0 \vee B_1 \ \ .
\end{aligned}
$$

Secondly, recall that

- conjunction is monotonic, that is, $(P_0 \wedge Q) \Rightarrow (P_1 \wedge Q)$ if $P_0 \Rightarrow P_1$;
- implication is anti-monotonic in its first argument, that is $(P_0 \Rightarrow Q) \Rightarrow (P_1 \Rightarrow Q)$ if $P_1 \Rightarrow P_0$.

Therefore we have

$$
\begin{aligned}
& wp\ (\textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1\ \textbf{fi})\ Q \\
= \ & (B_0 \Rightarrow wp\ S_0\ Q) \wedge (B_1 \Rightarrow wp\ S_1\ Q) \wedge (B_0 \vee B_1) \\
= \quad & \{ \text{ since } B_0 \vee (B_1 \wedge \neg B_0) = B_0 \vee B_1 \ \} \\
& (B_0 \Rightarrow wp\ S_0\ Q) \wedge (B_1 \Rightarrow wp\ S_1\ Q) \wedge (B_0 \vee (B_1 \wedge \neg B_0)) \\
\Rightarrow \quad & \{ \text{ since } B_1 \wedge \neg B_0 \Rightarrow B_1, \text{(anti-)monotonicity as discussed above. } \} \\
& (B_0 \Rightarrow wp\ S_0\ Q) \wedge (B_1 \wedge \neg B_0 \Rightarrow wp\ S_1\ Q) \wedge (B_0 \vee (B_1 \wedge \neg B_0)) \\
= \ & wp\ (\textbf{if}\ B_0 \rightarrow S_0 \mid B_1 \wedge \neg B_0 \rightarrow S_1\ \textbf{fi})\ Q \ \ .
\end{aligned}
$$

Properties of Weakest Precondition

13. Prove that $(wp\ S\ Q_0 \vee wp\ S\ Q_1) \Rightarrow wp\ S\ (Q_0 \vee Q_1)$.

Solution: Recall from propositional logic that $(P \lor Q) \Rightarrow R$ iff. $(P \Rightarrow R) \land (Q \Rightarrow R)$.

$$
\begin{array}{ll}
& (wp\ S\ Q_0 \lor wp\ S\ Q_1) \Rightarrow wp\ S\ (Q_0 \lor Q_1) \\
\equiv & \quad \{ \text{ said property above } \} \\
& (wp\ S\ Q_0 \Rightarrow wp\ S\ (Q_0 \lor Q_1)) \land \\
& (wp\ S\ Q_1 \Rightarrow wp\ S\ (Q_0 \lor Q_1)) \\
\Leftarrow & \quad \{ \text{ Monotonicity } \} \\
& (Q_0 \Rightarrow (Q_0 \lor Q_1)) \land (Q_1 \Rightarrow (Q_0 \lor Q_1)) \\
\equiv & \mathit{True} \ .
\end{array}
$$

14. Recall the definition of Hoare triple in terms of *wp*:

$$\{P\}\, S\, \{Q\} \ = \ P \Rightarrow wp\ S\ Q \ .$$

Prove that

1. $(\{P\}\, S\, \{Q\} \land (P_0 \Rightarrow P)) \Rightarrow \{P_0\}\, S\, \{Q\}$.
2. $\{P\}\, S\, \{Q\} \land \{P\}\, S\, \{R\} \equiv \{P\}\, S\, \{Q \land R\}$.

Solution:

1. We reason:

$$
\begin{array}{ll}
& \{P_0\}\, S\, \{Q\} \\
\equiv & \quad \{ \text{ definition of Hoare triple } \} \\
& P_0 \Rightarrow wp\ S\ Q \\
\Leftarrow & \quad \{ \text{ since } P_0 \Rightarrow P \ \} \\
& P \Rightarrow wp\ S\ Q \\
\equiv & \quad \{ \text{ definition of Hoare triple } \} \\
& \{P\}\, S\, \{Q\} \ .
\end{array}
$$

2. We reason:

$$
\begin{array}{ll}
& \{P\}\, S\, \{Q \land R\} \\
\equiv & \quad \{ \text{ definition of Hoare triple } \} \\
& P \Rightarrow wp\ S\ (Q \land R) \\
\equiv & \quad \{ \text{ distributivity over conjunction } \} \\
& P \Rightarrow (wp\ S\ Q \land wp\ S\ R) \\
\equiv & \quad \{ \text{ since } (P \Rightarrow (X \land Y)) \equiv (P \Rightarrow X) \land (P \Rightarrow Y) \ \} \\
& (P \Rightarrow wp\ S\ Q) \land (P \Rightarrow wp\ S\ R) \\
\equiv & \quad \{ \text{ definition of Hoare triple } \} \\
& \{P\}\, S\, \{Q\} \land \{P\}\, S\, \{R\} \ .
\end{array}
$$

15. Recall the weakest precondition of **if**:

$$wp\ (\textbf{if}\ B_0 \to S_0 \mid B_1 \to S_1\ \textbf{fi})\ Q \ = \ (B_0 \Rightarrow wp\ S_0\ Q) \land (B_1 \Rightarrow wp\ S_1\ Q) \land (B_0 \lor B_1) \ .$$

Prove that

$$\{P\}\,\mathbf{if}\ B_0 \to S_0 \mid B_1 \to S_1\ \mathbf{fi}\,\{Q\} \ \equiv$$
$$\quad \{P \wedge B_0\}\,S\,\{Q\} \ \wedge\ \{P \wedge B_1\}\,S\,\{Q\} \ \wedge\ (P \Rightarrow (B_0 \vee B_1)) \ .$$

Note: having proved so shows that the way we annotate **if** is correct:

$$\begin{array}{l}
\{P\} \\
\mathbf{if}\ B_0 \to \{P \wedge B_0\}\,S_0\,\{Q\} \\
\ \mid\ B_1 \to \{P \wedge B_1\}\,S_1\,\{Q\} \\
\mathbf{fi} \\
\{Q\} \ .
\end{array}$$

Solution: We reason:

$$\begin{array}{ll}
 & \{P\}\,\mathbf{if}\ B_0 \to S_0 \mid B_1 \to S_1\ \mathbf{fi}\,\{Q\} \\
= & \quad \{ \text{definition of Hoare triple} \} \\
 & P \Rightarrow wp\ (\mathbf{if}\ B_0 \to S_0 \mid B_1 \to S_1\ \mathbf{fi})\ Q \\
= & \quad \{ \text{definition of } wp \} \\
 & P \Rightarrow ((B_0 \Rightarrow wp\ S_0\ Q) \wedge (B_1 \Rightarrow wp\ S_1\ Q) \wedge (B_0 \vee B_1)) \\
= & \quad \{ \text{since } (P \Rightarrow (Q \wedge R)) \equiv (P \Rightarrow Q) \wedge (P \Rightarrow R) \} \\
 & (P \Rightarrow (B_0 \Rightarrow wp\ S_0\ Q)) \wedge \\
 & (P \Rightarrow (B_1 \Rightarrow wp\ S_1\ Q)) \wedge \\
 & (P \Rightarrow (B_0 \vee B_1)) \\
= & \quad \{ \text{since } (P \Rightarrow (Q \Rightarrow R)) \equiv ((P \wedge Q) \Rightarrow R) \} \\
 & ((P \wedge B_0) \Rightarrow wp\ S_0\ Q) \wedge \\
 & ((P \wedge B_1) \Rightarrow wp\ S_1\ Q) \wedge \\
 & (P \Rightarrow (B_0 \vee B_1)) \\
= & \quad \{ \text{definition of Hoare triple} \} \\
 & \{P \wedge B_0\}\,S_0\,\{Q\} \wedge \\
 & \{P \wedge B_1\}\,S_1\,\{Q\} \wedge \\
 & (P \Rightarrow (B_0 \vee B_1)) \ .
\end{array}$$

16. Recall that $wp\ S\ Q$ stands for "the weakest precondition for program S to terminate in a state satisfying Q". What programs S, if any, satisfy each of the following conditions?

    1. $wp\ S\ True = True$.
    2. $wp\ S\ True = False$.
    3. $wp\ S\ False = True$.
    4. $wp\ S\ False = False$.

Solution:

1. $wp\ S\ True = True$: S is a program that always terminates.
2. $wp\ S\ True = False$: S is a program that never terminates.
3. $wp\ S\ False = True$: there is no such a program S.
4. $wp\ S\ False = False$: S can be any program.