Solutions to demonstration problems

Solution to Problem 4

Boolean statements can be represented using basic cases, thus

\[ a == b \equiv \text{def} \neg (a > b) \land \neg (b > a) \]

and

\[ a < b \equiv \text{def} \quad b > a \]
\[ a \neq b \equiv \text{def} \quad \neg (a == b) \]

We choose \( A = "a > b" \) and \( B = "b > a" \) as atomic propositions. This way the statement in item (a) is

\[ \neg ((\neg A \land \neg B) \lor B) \]

and respectively, in item (b):

\[ \neg (\neg A \land \neg B) \land \neg B \]

Notice that the second proposition is obtained from the first by applying de Morgan’s rule and thus the statements are logically equivalent.

Solution to Problem 5

(a) \( \models_p [x > 0] y = x + 1 \ [y > 1] \)

Starting from the postcondition and applying the rule for assignment backwards, we obtain \([x + 1 > 1] y = x + 1 \ [y > 1]\)

\( x > 0 \) is equivalent to \( x + 1 > 1 \), and the claim holds.

(b) \( \models_p [\text{true}] y = x \ ; \ y = x + x + y \ [y == 3 \times x] \)

Applying twice the assignment rule, we obtain:

\([x + x + y == 3 \times x] y = x + x + y \ [y == 3 \times x]\)
\([x + x + x == 3 \times x] y = x \ [x + x + y == 3 \times x]\)
and furthermore using the rule for composition:

\[ x + x + x = 3x \]
\[ y = x ; \ y = x + x + y \]
\[ y = 3x \].

Statement \( x + x + x = 3x \) evaluates to true for all integers, and thus the claim holds.

(c) \( \models_p [x > 1] a = 1 ; y = x ; y = y - a \ y > 0 \ & \ x > y \)

\[ y - a > 0 \ & \ x > y - a \ y = y - a \ y > 0 \ & \ x > y \]
\[ x - a > 0 \ & \ x > x - a \ y = x \ y - a > 0 \ & \ x > y - a \]
\[ x - 1 > 0 \ & \ x > x - 1 \ a = 1 \ x - a > 0 \ & \ x > x - a \].

Now, the latter part of \( x - 1 > 0 \ & \ x > x - 1 \) evaluates to true for all integers and \( x - 1 > 0 \) is equivalent to \( x > 1 \). Thus the claim holds.

**Solution to Problem 6**

\[ \text{true} \ & \ ! (x > y) \ [! (x > y) \ [x <= y] \ [x = min(x, y)] \ z = x \ [z = min(x, y)] \text{ and} \]
\[ \text{true} \ & \ (x > y) \ [(x > y)] \ [y = min(x, y)] \ z = y \ [z = min(x, y)] \].

Thus,

\[ \text{true} \]
\[ \text{if}(x > y) \ \text{then} \{ \]
\[ \quad \ z = y \]
\[ \} \ \text{else} \{ \]
\[ \quad \ z = x \]
\[ \} \ [z = min(x, y)] \]

**Solution to Problem 7**

First, we need and invariant for the loop. Inspecting the code, we note that the value of variable \( z \) increases while the value for variable \( v \) decreases. Moreover, the sum of \( z \) and \( v \) stays constant. This constant is obtained for the initial values of \( z \) and \( v \), as thus we have invariant \( I: z + v = x + y \).

We check that \( I \) really is an invariant:

\[ z + v - 1 = x + y \ v = v - 1 \ [z + v = x + y] \]
\[ z + v = x + y \ [z + v - 1 = x + y] \ z = z + 1 \ [z + v - 1 = x + y] \]
Thus:

\[
\begin{align*}
& z + v = x + y \\
& \text{while}(!(v == 0)) \{
& \quad z = z + 1; \\
& \quad v = v - 1 \\
& \}\ [z + v = x + y \ \& \& v = 0]
\end{align*}
\]

Finally, we need to find the preconditions for the assignments before the loop:

\[
\begin{align*}
& z + y = x + y \\
& v = y \\
& x + y = x + y \\
& z = x \\
& z + y = x + y
\end{align*}
\]

Now \( x + y = x + y \) evaluates to true for all integers.

(b) \( \models_t [0 \leq y] \ \sum \ [z = x + y] \)

To prove total correctness, we need to make sure the program terminates.

\[
\begin{align*}
& [0 \leq y] [x + y = x + y \ \& \& 0 \leq y] z = x [z + y = x + y \ \& \& 0 \leq y] \\
& [z + y = x + y \ \& \& 0 \leq y] v = y [z + v = x + y \ \& \& 0 \leq v] \\
& \text{while}(!(v == 0)) \{
& \quad [z + v = x + y \ \& \& 0 \leq v - 1 \ \& \& v - 1 < n] z = z + 1; \\
& \quad [z + v - 1 = x + y \ \& \& 0 \leq v - 1 \ \& \& v - 1 < n] \\
& \quad [z + v - 1 = x + y \ \& \& 0 \leq v - 1 \ \& \& v - 1 < n] v = v - 1 \\
& \quad [z + v = x + y \ \& \& 0 \leq v \ \& \& v < n] \\
& \}\ [z + v = x + y \ \& \& 0 \leq v \ \& \& v = 0] [z = x + y]
\end{align*}
\]