CS245 Tutorial

Nov. 23, 2015

1. Show that the following Hoare triple is satisfied under partial correctness.

   $(\!| x = n |\!)$
   `x = x + 1;`
   $(\!| x = n + 1 |\!)$

   Solution: Working backwards from the postcondition we can prove using the assignment inference rule the following specification:

   $(\!| x = n |\!)$
   $(\!| x + 1 = n + 1 |\!)$ implied (a)
   `x = x + 1;`
   $(\!| x = n + 1 |\!)$ assignment

   It remains to supply the implied proof (a): $(x = n) \rightarrow (x+1 = n+1)$. Clearly this is true by a bit of algebraic manipulation. (A more detailed proof from first principles could be given using the ideas presented in the Peano Arithmetic lectures.)

2. Show that the following Hoare triple is satisfied under partial correctness.

   $(\!| x \geq 0 |\!)$
   `y = 0;`
   $(\!| x + y \geq 0 |\!)$

   Solution: Working backwards from the postcondition we can prove using the assignment inference rule the following specification:

   $(\!| x \geq 0 |\!)$
   $(\!| x + 0 \geq 0 |\!)$ implied (a)
   `y = 0;`
   $(\!| x + y \geq 0 |\!)$ assignment

   It remains to supply the implied proof (a): $(x \geq 0) \rightarrow (x + 0 \geq 0)$. Clearly this is true by a bit of algebraic manipulation. (A more detailed proof from first principles could be given using the ideas presented in the Peano Arithmetic lectures.)

3. Find a precondition, such that the following Hoare triple is satisfied under partial correctness.
   $(\!|\ ???\ |\!)$
   `x = x + 1;`
   `y = y + 2;`
   $(\!|\ (x = y)\ |\!)$

   Solution:

   $(\!|\ (x + 1 = y + 2)\ |\!)$
   `x = x + 1;`
   $(\!|\ (x = y + 2)\ |\!)$ assignment
   `y = y + 2;`
   $(\!|\ (x = y)\ |\!)$ assignment

   So a suitable precondition to make the Hoare triple satisfied under partial correctness is $(x + 1 = y + 2)$.

4. Show that the following Hoare triple is satisfied under partial correctness.

$(\!|\ \mathit{true}\ |\!)$
```
if (x > 0) {
    y = x ;
} else {
    y = -x ;
}
```
$(\!|\ (x > 0 \wedge y = x) \vee (\neg(x > 0) \wedge y = -x)\ |\!)$

Solution:

Annotated Program:

$(\!|\ \mathit{true}\ |\!)$
```
if (x > 0) {
```
$(\!|\ (x > 0)\ |\!)$	if-then-else
$(\!|\ (x > 0 \wedge x = x) \vee (\neg(x > 0) \wedge x = -x)\ |\!)$	implied (a)
`y = x;`	
$(\!|\ (x > 0 \wedge y = x) \vee (\neg(x > 0) \wedge y = -x)\ |\!)$	assignment

```
} else {
```
$(\!|\ \neg(x > 0)\ |\!)$	if-then-else
$(\!|\ (x > 0 \wedge -x = x) \vee (\neg(x > 0) \wedge -x = -x)\ |\!)$	implied (b)
`y = -x;`	
$(\!|\ (x > 0 \wedge y = x) \vee (\neg(x > 0) \wedge y = -x)\ |\!)$	assignment

```
}
```
$(\!|\ (x > 0 \wedge y = x) \vee (\neg(x > 0) \wedge y = -x)\ |\!)$ if-then-else

Proof of implied (a): $(x > 0) \rightarrow ((x > 0 \wedge x = x) \vee (\neg(x > 0) \wedge x = -x))$:

1.	$x > 0$	assumption
2.	$x = x$	$=$i
3.	$(x > 0) \wedge (x = x)$	$\wedge$i : 1, 2
4.	$((x > 0) \wedge x = x) \vee (\neg(x > 0) \wedge x = -x)$	$\vee$i : 3
5.	(Line 1) $\rightarrow$ (Line 4)	$\rightarrow$i : 1–4

Proof of implied (b): Similar.

5. Show that the following Hoare triple is satisfied under partial correctness.

$(\!|\ \mathit{true}\ |\!)$
```
if (x > y) {
    m = x + 2 * y;
} else if (x < y) {
    m = x + 3 * y;
} else {
    m = x + 4 * y;
}
```
$(\!|\ (x > y \wedge m = x + 2y) \vee (x < y \wedge m = x + 3y) \vee (x = y \wedge m = x + 4y)\ |\!)$

Solution:

Let $s(m, x, y)$ denote

$$(x > y \wedge m = x + 2y) \vee (x < y \wedge m = x + 3y) \vee (x = y \wedge m = x + 4y).$$

Annotated Program:

$(\!|\ \mathit{true}\ |\!)$	
`if (x > y) {`	
$(\!|\ (x > y)\ |\!)$	if-then-else
$(\!|\ s(x + 2y, x, y)\ |\!)$	implied (a)
`m = x + 2 * y;`	
$(\!|\ s(m, x, y)\ |\!)$	assignment
`} else if (x < y) {`	
$(\!|\ ((x < y) \wedge (\neg(x > y)))\ |\!)$	if-then-else
$(\!|\ s(x + 3y, x, y)\ |\!)$	implied (b)
`m = x + 3 * y;`	
$(\!|\ s(m, x, y)\ |\!)$	assignment
`} else {`	
$(\!|\ ((\neg(x < y)) \wedge (\neg(x > y)))\ |\!)$	if-then-else
$(\!|\ s(x + 4y, x, y)\ |\!)$	implied (c)
`m = x + 4 * y;`	
$(\!|\ s(m, x, y)\ |\!)$	assignment
`}`	
$(\!|\ s(m, x, y)\ |\!)$	if-then-else

Proof of implied (a): $(x > y) \to s(x + 2y, x, y)$:

1.	$x > y$	assumption
2.	$x + 2y = x + 2y$	=i
3.	$(x > y) \wedge (x + 2y = x + 2y)$	$\wedge$i : 1, 2
4.	$((x > y) \wedge (x + 2y = x + 2y)) \vee$ $((x < y) \wedge (x + 2y = x + 3y)) \vee$ $((x = y) \wedge (x + 2y = x + 4y))$	$\vee$i : 3
5.	$(x > y) \to$ $(((x > y) \wedge (x + 2y = x + 2y)) \vee$ $((x < y) \wedge (x + 2y = x + 3y)) \vee$ $((x = y) \wedge (x + 2y = x + 4y)))$	$\to$i : 1–4

Proof of implied (b): $(\neg(x > y) \wedge (x < y)) \to s(x + 3y, x, y)$:

1.	$\neg(x > y) \wedge (x < y)$	assumption
2.	$x < y$	$\wedge$e : 1
3.	$x + 3y = x + 3y$	=i
4.	$(x < y) \wedge (x + 3y = x + 3y)$	$\wedge$i : 2, 3
5.	$((x > y) \wedge (x + 3y = x + 2y)) \vee$ $((x < y) \wedge (x + 3y = x + 3y)) \vee$ $((x = y) \wedge (x + 3y = x + 4y))$	$\vee$i : 3
6.	$(\neg(x > y) \wedge (x < y)) \to$ $(((x > y) \wedge (x + 3y = x + 2y)) \vee$ $((x < y) \wedge (x + 3y = x + 3y)) \vee$ $((x = y) \wedge (x + 3y = x + 4y)))$	$\to$i : 1–5

Proof of implied (c): $(\neg(x > y) \wedge \neg(x < y)) \to s(x + 4y, x, y)$:

1.	$\neg(x > y) \wedge \neg(x < y)$	assumption
2.	$x = y$	algebra : 1
3.	$x + 4y = x + 4y$	=i
4.	$(x = y) \wedge (x + 4y = x + 4y)$	$\wedge$i : 2, 3
5.	$(((x > y) \wedge (x + 4y = x + 2y)) \vee$ $((x < y) \wedge (x + 4y = x + 3y)) \vee$ $((x = y) \wedge (x + 4y = x + 4y)))$	$\vee$i : 3
6.	$(\neg(x > y) \wedge \neg(x < y)) \to$ $(((x > y) \wedge (x + 4y = x + 2y)) \vee$ $((x < y) \wedge (x + 4y = x + 3y)) \vee$ $((x = y) \wedge (x + 4y = x + 4y)))$	$\to$i : 1–5

6. Consider the following incomplete specification:

$(\!| (x = x_0) \wedge (y = y_0) \wedge (???) |\!)$

```
while (x != 0 || y != 0) {
    if (x > 0) {
        x = x - 1 ;
    } else {
        x = y ;
        y = y - 1 ;
    }
}
```

$(\!| ??? |\!)$

- The post-condition of the program is: (choose the answer)
  (a) $x + y = 0$
  (b) $x + y = \sum_{i=1}^{x_0} i$
  (c) $x - y = \sum_{i=1}^{y_0} i$
  (d) $x + y > 0$
- A pre-condition for the termination of the program is: (choose the answer)
  (a) $x = x_0 \wedge y = y_0 \wedge x = 0 \wedge y \geq 0$
  (b) $x = x_0 \wedge y = y_0 \wedge x \geq 0 \wedge y \geq 0$
  (c) $x = x_0 \wedge y = y_0 \wedge y \geq 0$
  (d) $x = x_0 \wedge y = y_0 \wedge x = 0 \wedge y = 0$

Solution:

- We observe that the program will terminate when $x = 0$ and $y = 0$ (the while condition). Thus the post-condition of the program is: $x + y = 0$.
- A pre-condition for the termination of the program is: $x = x_0 \wedge y = y_0 \wedge y \geq 0$. All of them could be a pre-condition, but the one that "includes" the others is $x = x_0 \wedge y = y_0 \wedge y \geq 0$.

7. Consider the following code and its pre- and post-conditions.

$(\!| \mathit{true} |\!)$

```
x = 0 ;
s = 0 ;
while ( x <= n ) {
    s = s + x ;
    x = x + 1 ;
}
```

$(\!| s = n(n+1)/2 |\!)$

(a) Choose a suitable loop invariant, and annotate the program.

Solution: The loop invariant $0 = (x-1)x/2 \wedge x \leq n+1$ yields the following annotations.

$(\!\| \mathit{true} \|\!)$	
$(\!\| 0 = (0-1)0/2 \wedge 0 \leq n+1 \|\!)$	implied (a)
`x = 0 ;`	
$(\!\| 0 = (x-1)x/2 \wedge x \leq n+1 \|\!)$	assignment
`s = 0 ;`	
$(\!\| s = (x-1)x/2 \wedge x \leq n+1 \|\!)$	assignment
`while ( x <= n ) {`	
$(\!\| s = (x-1)x/2 \wedge x \leq n+1 \wedge x \leq n \|\!)$	partial-`while`
$(\!\| s + x = x(x+1)/2 \wedge x+1 \leq n+1 \|\!)$	implied (b)
`s = s + x ;`	
$(\!\| s = x(x+1)/2 \wedge x+1 \leq n+1 \|\!)$	assignment
`x = x + 1 ;`	
$(\!\| s = (x-1)x/2 \wedge x \leq n+1 \|\!)$	assignment
`}`	
$(\!\| s = (x-1)x/2 \wedge x \leq n+1 \wedge x \not\leq n \|\!)$	partial-`while`
$(\!\| s = n(n+1)/2 \|\!)$	implied (c)

(b) Complete the proof of partial correctness.

Solution:
Implied (a) is $\mathit{true} \rightarrow (0 = (0-1)0/2 \wedge 0 \leq n+1)$: Trivial arithmetic.
Implied (b) is $(s = (x-1)x/2 \wedge x \leq n+1 \wedge x \leq n) \rightarrow$
$(s + x = x(x+1)/2 \wedge x+1 \leq n+1)$:

The premise yields

$$s + x = \frac{(x-1)x}{2} + x = \frac{x^2 - x}{2} + \frac{2x}{2} = \frac{x^2 + x}{2} = \frac{x(x+1)}{2} \ ,$$

as required.

Implied (c) is $(s = (x-1)x/2 \wedge x \leq n+1 \wedge x \not\leq n) \rightarrow (s = n(n+1)/2)$: Trivial arithmetic.

(c) Explain why the code and conditions satisfy total correctness. As part of your explanation, choose a suitable variant expression.

Solution: One suitable variant expression is $n - x$. At the start of the `while`-loop, it is non-negative. The first assignment of the loop leaves $n - x$ unchanged, and the second assignment decreases it. Thus $n - x$ eventually becomes negative and the loop terminates.

Together with the partial correctness already proven, this implies total correctness.

8. Show that the following Hoare triple is satisfied under total correctness. The code computes the quotient and the remainder when dividing x by y.

$(\!| (x \geq 0) \wedge (y \geq 0) |\!)$
```
q = 0 ;
r = x ;
while (r >= y ) {
    r = r - y ;
    q = q + 1 ;
}
```
$(\!| (x = q \cdot y + r) \wedge (r \geq 0) \wedge (r < y) |\!)$

Solution: For our loop invariant I, we select the formula

$$(x = q \cdot y + r) \wedge (r \geq 0).$$

With this choice, we obtain the following annotated program.

$(\!| (x \geq 0) \wedge (y \geq 0) |\!)$	
$(\!| (x = 0 \cdot y + x) \wedge (x \geq 0) |\!)$	implied (a)
`q = 0 ;`	
$(\!| (x = q \cdot y + x) \wedge (x \geq 0) |\!)$	assignment
`r = x ;`	
$(\!| (x = q \cdot y + r) \wedge (r \geq 0) |\!)$	assignment
`while (r >= y ) {`	
$\quad (\!| (x = q \cdot y + r) \wedge (r \geq 0) \wedge (r \geq y) |\!)$	partial-while
$\quad (\!| (x = (q+1) \cdot y + (r - y)) \wedge ((r - y) \geq 0) |\!)$	implied (b)
`  r = r - y ;`	
$\quad (\!| (x = (q+1) \cdot y + r) \wedge (r \geq 0) |\!)$	assignment
`  q = q + 1 ;`	
$\quad (\!| (x = q \cdot y + r) \wedge (r \geq 0) |\!)$	assignment
`}`	
$(\!| (x = q \cdot y + r) \wedge (r \geq 0) \wedge \neg(r \geq y) |\!)$	partial-while
$(\!| (x = q \cdot y + r) \wedge (r \geq 0) \wedge (r < y) |\!)$	implied (c)

To complete the proof of partial correctness, we must prove the following.

- Implied (a): $((x \geq 0) \wedge (y \geq 0)) \rightarrow ((x = 0 \cdot y + x) \wedge (x \geq 0))$
- Implied (b): $((x = q \cdot y + r) \wedge (r \geq 0) \wedge (r \geq y)) \rightarrow$
  $((x = (q+1) \cdot y + (r - y)) \wedge ((r - y) \geq 0))$
- Implied (c): $((x = q \cdot y + r) \wedge (r \geq 0) \wedge \neg(r \geq y)) \rightarrow ((x = q \cdot y + r) \wedge (r \geq 0) \wedge (r < y))$

All of these can be proven with simple algebraic manipulation. This establishes that the given Hoare triple is satisfied under partial correctness.

To show that it is satisfied under total correctness, we must prove that the program always terminates. Consider the candidate loop variant: r.

- Initially, $x \geq 0$ and $r = x$, so r is non-negative.
- Each time through the loop, x does not change and r is decreased by y. Initially, $y \geq 0$ and y does not change; hence r decreases each time through the body of the while loop.
- The guard $r \geq y$ prevents $r < 0$, so the variant stays non-negative.
- Hence our candidate loop variant has the required properties to be a loop variant. This completes the proof of satisfaction under total correctness.