

Programming Languages: Imperative Program Construction

Practicals 0: Non-Looping Constructs and Weakest Precondition

Shin-Cheng Mu

Autumn Term, 2021

Guarded Command Language Basics

1. Which of the following Hoare triples hold?

- (a) $\{x = 7\} \text{skip} \{ \text{odd } x \};$
- (b) $\{x > 60\} x := x \times 2 \{x > 100\};$
- (c) $\{x > 40\} x := x \times 2 \{x > 100\};$
- (d) $\{ \text{true} \} \text{if } x \leq y \rightarrow y := y - x \mid x \geq y \rightarrow x := x - y \text{ fi} \{x \geq 0 \wedge y \geq 0\};$
- (e) $\{ \text{even } x \wedge \text{even } y \} \text{if } x \leq y \rightarrow y := y - x \mid x \geq y \rightarrow x := x - y \text{ fi} \{ \text{even } x \wedge \text{even } y \}.$

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

3. Verify:

```
{x = X ∧ y = Y}
x := x ≠ y
y := x ≠ y
x := x ≠ y
{x = Y ∧ y = X}
```

where x and y are boolean and (\neq) 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 :

associative : $a \otimes (b \otimes c) = (a \otimes b) \otimes c$,
unipotent : $a \otimes a = 1$,

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

4. Verify the following program:

```
var r, b : Int
{0 ≤ r < 2 × b}
if b ≤ r → r := r - b
| r < b → skip
fi
{0 ≤ r < b}
```

5. Verify:

```

var  $x, y : Int$ 
 $\{True\}$ 
 $x, y := x \times x, y \times y$ 
if  $x \geq y \rightarrow x := x - y$ 
     $| y \geq x \rightarrow y := y - x$ 
fi
 $\{x \geq 0 \wedge y \geq 0\}$  .

```

6. Verify:

```

var  $a, b : Bool$ 
 $\{True\}$ 
if  $\neg a \vee b \rightarrow a := \neg a$ 
     $| a \vee \neg b \rightarrow b := \neg b$ 
fi
 $\{a \vee b\}$  .

```

Weakest Precondition

7. 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) $x := x \times 2, \text{even } x$;
- (c) $x := x \times 2, x > 100 \wedge \text{even } x$;
- (d) $x := x \times 2, \text{odd } x$.
- (e) $\text{skip}, \text{odd } x$.

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

9. Recall the definition of Hoare triple in terms of wp :

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

Prove that

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

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

$$wp\ (\text{if } B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1 \text{ fi})\ Q = (B_0 \Rightarrow wp\ S_0\ Q) \wedge (B_1 \Rightarrow wp\ S_1\ Q) \wedge (B_0 \vee B_1) .$$

Prove that

$$\{P\} \text{if } B_0 \rightarrow S_0 \mid B_1 \rightarrow S_1 \text{ fi } \{Q\} \equiv \{P \wedge B_0\} S \{Q\} \wedge \{P \wedge B_1\} S \{Q\} \wedge (P \Rightarrow (B_0 \vee B_1)) .$$

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

```

 $\{P\}$ 
if  $B_0 \rightarrow \{P \wedge B_0\} S_0 \{Q\}$ 
     $| B_1 \rightarrow \{P \wedge B_1\} S_1 \{Q\}$ 
fi
 $\{Q\}$  .

```

11. 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. $wp\ S\ False = False$.