Department of Computer Science, Australian National University
COMP2600 — Formal Methods in Software Engineering
Semester 2, 2010

Week 5 Tutorial

Hoare Logic

1 Hoare Notation — Warm-up Exercises

These questions are meant to foster the feeling that Hoare notation is a compact way of saying things that you want to say when discussing a program. Each of the following parts consists of a statement in Hoare Logic — each is a simple assertion about a piece of code. In each case you should just say whether the statement is true or false.

All that is wanted for this group of problems is just a true-or-false answer based on your intuitions. Can your intuitions be programmed, though?

(The program variables in this and later questions are typed integer.)

a) $\{j = a\}$ `j:=j+1` $\{a = j + 1\}$

b) $\{i = j\}$ `i:=j+i` $\{i > j\}$

c) $\{j = a + b\}$ `i:=b; j:=a` $\{j = 2 * a\}$

d) $\{i > j\}$ `j:=i+1; i:=j+1` $\{i > j\}$

e) $\{i \neq j\}$ `if i>j then m:=i-j else m:=j-i` $\{m > 0\}$

f) $\{i = 3 * j\}$ `if i>j then m:=i-j else m:=j-i` $\{m - 2 * j = 0\}$

g) $\{x = b\}$ `while x>a do x:=x-1` $\{b = a\}$

2 Hoare Proof Rules I — Assignment Statements

Next up, we will be focusing on the natural deduction rules for precondition strengthening, postcondition weakening and especially the use of the assignment axiom.

a) Prove $\{i = 5\}$ `a:=i+2` $\{(a = 7) \wedge (i = 5)\}$

b) Prove $\{i = 5\}$ `a:=i+2` $\{a = 7\}$

c) Prove $\{i = 5\}$ `a:=i+2` $\{(a = 7) \wedge (i > 0)\}$

d) Prove $\{(i = 5) \wedge (a = 3)\}$ `a:=i+2` $\{a = 7\}$

e) Prove $\{a = 7\}$ `i:=i+2` $\{a = 7\}$

f) Prove $\{i = a - 1\}$ `i:=i+2` $\{i = a + 1\}$

g) Prove $\{i = 5\}$ `i:=i+2` $\{i > 0\}$

h) Prove $\{True\}$ `a:=i+2` $\{a = i + 2\}$

3 Hoare Proof Rules for Control Structures

Remember the way proofs of larger program fragments are meant to be constructed. If your immediate goal is to prove some property of a conditional statement, or a loop or a sequence of statements, then you use the corresponding Hoare Rule to generate appropriate subgoals.

3.1 Sequencing

a) Prove $\{a > b\}$ `m:=1; n:=a-b` $\{m * n > 0\}$

b) Prove $\{s = 2^i\}$ `i:=i+1; s:=s*2` $\{s = 2^i\}$

3.2 Conditionals

c) Prove $\{True\}$ `if i<j then min:=i else min:=j` $\{(min \leq i) \wedge (min \leq j)\}$

d) Prove $\{i > 0 \wedge j > 0\}$ `if i<j then min:=i else min:=j` $\{min > 0\}$

3.3 Loops

e) Prove $\{s = 2^i\}$ `while i<n do i:=i+1; s:=s*2` $\{s = 2^i\}$

4 Want More?

You can get a bit more practice with the warm-up questions. If they were statements that were intuitively true then you should be able to prove them. Otherwise, beat them into shape and prove the better version.

5 Appendix: Hoare Logic Rules

- Precondition Strengthening:

$$\frac{\{P_w\}\ S\ \{Q\} \qquad P_s \implies P_w}{\{P_s\}\ S\ \{Q\}}$$

- Postcondition Weakening:

$$\frac{\{P\}\ S\ \{Q_s\} \qquad Q_s \implies Q_w}{\{P\}\ S\ \{Q_w\}}$$

- Assignment:

$$\{Q(e)\}\ \texttt{x} := \texttt{e}\ \{Q(x)\}$$

- Sequence:

$$\frac{\{P\}\ S_1\ \{Q\} \qquad \{Q\}\ S_2\ \{R\}}{\{P\}\ S_1; S_2\ \{R\}}$$

- Conditional:

$$\frac{\{P \wedge b\}\ S_1\ \{Q\} \qquad \{P \wedge \sim b\}\ S_2\ \{Q\}}{\{P\}\ \textbf{if}\ b\ \textbf{then}\ S_1\ \textbf{else}\ S_2\ \{Q\}}$$

- While Loop:

$$\frac{\{P \wedge b\}\ S\ \{P\}}{\{P\}\ \textbf{while}\ b\ \textbf{do}\ S\ \{P \wedge \sim b\}}$$