IIT CS536: Science of Programming

Homework 3: Hoare triples and proofs

Prof. Stefan Muller

Out: Tuesday, Sept. 26
Due: Monday, Oct. 9, 11:59pm CDT

Updated Oct. 4

This assignment contains 10 written task(s) for a total of 77 points.

Logistics

Submission Instructions

Please read and follow these instructions carefully.

- Submit your homework on Blackboard under the correct assignment by the deadline (or the extended deadline if taking late days).
- You may submit multiple times, but we will only look at your last submission. Make sure your last submission contains all necessary files.
- Email the instructor and TAs ASAP if
  - You submit before the deadline but then decide to take (more) late days.
  - You accidentally resubmit after the deadline, but did not intend to take late days.

  Otherwise, you do not need to let us know if you're using late days; we'll count them based on the date of your last submission.
- Submit your written answers in a single PDF or Word document. Typed answers are preferred (You can use any program as long as you can export a .pdf, .doc or .docx; LaTeX is especially good for typesetting logic and math, and well worth the time to learn it), but *legible* handwritten and scanned answers are acceptable as well.
- Your Blackboard submission should contain only the file with your written answers. Do not compress or put any files in folders.

Collaboration and Academic Honesty

This homework is to be completed individually. Read the policy on the website and be sure you understand it.

1 Hoare triples

Task 1.1 (Written, 10 points).

Let $s = \textsf{while } i < x \textsf{ do } x := x * i; i := i + 1 \textsf{ od}$. For each of the following, is the triple satisfied or unsatisfied in the given state? Explain why by unfolding the definition as we saw in the class. Note that some of these are partial correctness triples and others are total correctness. *Updated 10/4*: Added k to state of part d

a) $\{i = 1, x = 6\} \vDash [\ i < x\ ]\ s\ [\ i = x\ ]$

b) $\{i = -1, x = 5\} \vDash \{i < x\}\ s\ \{i \geq 0 \wedge x \leq 0\}$

c) $\{i = 1, x = 0\} \vDash \{i < x\}\ s\ \{i = x\}$

d) $\{i = 1, x = 2, k = 2\} \vDash \{x = k\}\ s\ \{x = k!\}$

e) $\{i = 1, x = 6\} \vDash \{T\}\ s\ \{\exists k. x = k!\}$

Task 1.2 (Written, 6 points).

Consider program s in the previous task. For each of the following triples, say if it's valid (satisfied in all states or not). If not, provide a counterexample and then fix either the precondition, or the postcondition or the statement to make the triple valid. Don't make your change trivial (that is, don't make the precondition a contradiction, the postcondition a tautology or the statement soemthing that always errors or diverges).

a) $\{T\}\ s\ \{x > 0\}$

b) $\{x = k\}\ s\ \{x = k\}$

c) $[\ i = 1 \wedge x = k \wedge x > 0\ ]\ s\ [\ i = k \wedge x = k!\ ]$

Task 1.3 (Written, 5 points).

Fill in an appropriate precondition such that the following triple is valid. Don't provide a trivial precondition (that is, don't make the precondition a contradiction). *Updated 10/4*: Added minus sign to postcondition

$$[\ \underline{\qquad}\ ]\ n := -m; \textsf{while } n \neq 0 \textsf{ do } r := r * -3; n := n - 1 \textsf{ od}\ [\ r = 3^{-m}\ ]$$

Task 1.4 (Written, 3 points).

Write a precondition P_1 such that the triple $\vDash [P_1]\, x := \mathit{sqrt}(x)/y\, [\mathbf{T}]$ is valid. Don't make your precondition trivial (a contradiction). **Explain by unfolding the definition of the triple as we saw in the class.**

Task 1.5 (Written, 8 points).

a) Write a program S_2 and a precondition P_2 such that $\nvDash [P_2]S_2[\mathbf{T}]$. **Explain by unfolding the definition of the triple as we saw in the class.**

b) Have you used a while clause in S_2? If yes, can you provide another S_2 this time without using the while clause? If no, can you provide another S_2 this time with using the while clause?

Task 1.6 (Written, 8 points).

a) Write a triple for the following program that is valid if and only if the program terminates when the initial value of x is greater than three.

$$\mathsf{while}\ x > 1\ \mathsf{do}\ \mathsf{if}\ \mathit{even}(x)\ \mathsf{then}\ x := 5x + 1\ \mathsf{else}\ x := x/2\ \mathsf{fi}\ \mathsf{od}$$

b) Write another triple for the above program which is valid if and only if the program does not terminate when the initial value of x is less than or equal to three.

2 Substitution

Task 2.1 (Written, 10 points).

Compute the given substitutions. Just substitute the expression for the value; you don't need to simplify anything further. Recall that you may need to perform α-conversions to avoid variable capture. **Show the intermediate steps of substitution when quantifiers are involved, as we did in class.**

a) $[y + 2/y]\exists z.\forall x.(x + y \geq z + y)$

b) $[y + 2/x]\exists z.\forall x.(x + y \geq z + y)$

c) $[x + 2/y]\exists z.\forall x.(x + y \geq z + y)$

d) $[z/x](x \geq z \to (\exists z.\forall x.x + y \geq z + y) \wedge y > z)$

e) $[z/x](x \geq z \to (\exists x.x + y \geq z + y) \wedge y > z)$

3 Proofs and Proof Outlines

Task 3.1 (Written, 20 points).

a) Write a program in IMP that given an array a of *size two*, returns its maximum element in variable m.

b) Write a partial correctness triple to state that your program does what is described in part (a).

c) Prove your partial correctness triple in Hoare logic using either proof trees or Hilbert-stlye proofs.

d) Write a proof outline for your partial correctness triple. You can use either rule for if that we discussed in class.

Task 3.2 (Written, 7 points).

Convert the following proof outline to a Hilbert-style proof. The remainder operator $x\%y$ returns the remainder when x is divided by y.

$$
\begin{array}{ll}
\{x \geq 0\} & \\
\mathsf{if}(x\%3 = \overline{0})\ \mathsf{then} & \{x \geq 0 \wedge x\%3 = 0\} \\
\quad s := x & \{s \geq 0 \wedge s\%3 = 0\} \Rightarrow \{s\%3 = 0\} \\
\mathsf{else} & \{x \geq 0 \wedge x\%3 \neq 0\} \\
\quad \mathsf{if}(x\%3 = \overline{1})\ \mathsf{then} & \{x \geq 0 \wedge x\%3 \neq 0 \wedge x\%3 = 1\} \Rightarrow \{x - 1 \geq 0 \wedge x - 1\%3 = 0\} \\
\qquad s := x - \overline{1} & \{s \geq 0 \wedge s\%3 = 0\} \Rightarrow \{s\%3 = 0\} \\
\quad \mathsf{else} & \{x \geq 0 \wedge x\%3 \neq 0 \wedge x\%3 \neq 1\} \Rightarrow \{x - 2 \geq 0 \wedge x - 2\%3 = 0\} \\
\qquad s := x - \overline{2} & \{s \geq 0 \wedge s\%3 = 0\} \Rightarrow \{s\%3 = 0\} \\
\quad \mathsf{fi} & \{s\%3 = 0\} \\
\mathsf{fi} & \{s\%3 = 0\}
\end{array}
$$

4 One more wrap-up question

Task 4.1 (Written, 0 points).

How long (approximately) did you spend on this homework, in total hours of actual working time? Your honest feedback will help us with future homeworks.