Correctness ("Hoare") Triples, pt. 1

CS 536: Science of Programming, Fall 2022

Sep 26: Added q. 16, 17

A. Why

- To specify a program's correctness, we need to know its precondition (what must be true before executing it) and its postcondition (what should be true after it).

B. Objectives

At the end of this practice you should be able to

- Recognize syntactically correct correctness triples.
- Say whether a correctness triple is satisfied, given information about whether the current state satisfies the precondition, whether the statement terminates, and if it does, whether the terminating state satisfies the postcondition.

C. Questions

For all the questions below, you can assume (unless otherwise said) that $\sigma \in \Sigma$, not $\Sigma_\perp$. (I.e., we're not trying to start run a program after an infinite loop or runtime failure.)

1. For a loop-free program without runtime errors, is there any difference between partial and total correctness?
2. Say we're given $\sigma \vDash \{x > 0\}\ S\ \{y > x\}$ for all σ and we're given a state τ where $\tau(x) = -3$. Do we know what S will do if we run in τ? Must it terminate (without a runtime error)? Must it produce a runtime error? Must it diverge? Must $y > x$ afterwards? How about $y \leq x$?
3. For which σ does $\sigma \vDash \{x > 1\}\ y := x*x\ \{y > x\}$ hold? Is this triple valid?
4. For which σ does $\sigma \vDash \{x > 0\}\ y := x*x\ \{y > x\}$ hold? Is this triple valid?
5. Under partial correctness, does $\vDash \{F\}\ S\ \{q\}$ hold for all σ, q, and S? What about $\vDash \{p\}\ S\ \{T\}$? Do these triples say anything interesting about S?
6. Repeat the previous question under total correctness: Does $\vDash_{tot} \{F\}\ S\ \{q\}$ always hold? Does $\vDash_{tot} \{p\}\ S\ \{T\}$? If they do, then do they say anything interesting about S?

For Problems 7 – 14, say for each statement whether it's true or false and give a brief explanation. (Just a sentence or two is fine.) Assume $\sigma \in \Sigma$. (Remember, if $\sigma \vDash$ any predicate or triple, then $\sigma \neq \perp$.) In general, programs might be deterministic or nondeterministic.

7. If $\sigma \vDash \{p\}\ S\ \{q\}$, then $\sigma \vDash p$.
8. If $\sigma \nvDash \{p\}\ S\ \{q\}$, then $\sigma \nvDash p$.

9. If $M(S, \sigma) \subseteq \{\bot_d, \bot_e\}$, then $\sigma \vDash \{p\}\ S\ \{q\}$.
10. If $\sigma \vDash p$ and $M(S, \sigma) \cap \{\bot_d, \bot_e\} \neq \varnothing$, then $\sigma \nvDash_{tot} \{p\}\ S\ \{q\}$.
11. If $\sigma \vDash \{p\}\ S\ \{q\}$ and $\sigma \vDash p$, then every state in $M(S, \sigma)$ either $\in \{\bot_d, \bot_e\}$ or satisfies q.
12. If $\sigma \vDash \{p\}\ S\ \{q\}$ and $\sigma \nvDash p$, then every state in $M(S, \sigma)$ is either $\in \{\bot_d, \bot_e\}$ or satisfies $\neg q$.
13. For nondeterministic S, if $\sigma \nvDash \{p\}\ S\ \{q\}$, then $\tau \vDash \neg q$ for some $\tau \in M(S, \sigma)$ but it's possible that for some other $\xi \in M(S, \sigma)$, we have $\xi \vDash q$.
14. For nondeterministic S, if $\sigma \nvDash_{tot} \{p\}\ S\ \{q\}$, if $\bot \notin M(S, \sigma)$, then $\tau \vDash \neg q$ for some $\tau \in M(S, \sigma)$ but it's possible that for other some other $\xi \in M(S, \sigma)$, we have $\xi \vDash q$.

15. Let $S \equiv x := x * x;\ y := y * y$ and let $\sigma(x) = \alpha$ and $\sigma(\xi) = \beta$. Verify that $\sigma \vDash_{tot} \{x > y > 0\}\ S\ \{x > y > 0\}$ as follows. First assume σ satisfies the precondition, then calculate $M(S, \sigma)$, and then verify that $M(S, \sigma) - \bot$ satisfies the postcondition.

[2022-09-26 - added]

16. What are the mostly trivial cases for partial and total correctness?
17. How do we symbolically write that total correctness is partial correctness plus termination?

Solution to Practice 8 (Hoare Triples, pt 1)

1. For a loop-free, failure-free program, there's no difference between partial and total correctness.
2. No to all the questions: The triple only tells us what will happen if the precondition is satisfied. Since τ doesn't satisfy the precondition, the triple doesn't say anything about what will happen when you run S in τ. S might diverge, it might cause an error. It might terminate in which case the final state might satisfy the postcondition or it might not.
3. All states satisfy the triple, so the triple is valid.
4. States with $x > 1$ set y appropriately and do satisfy the triple. States with $x < 1$ satisfy the triple trivially. But in states with $x = 1$ do not satisfy the triple. So, the triple is not valid, since it's not satisfied in all states.
5. Under partial correctness, for all S, both triples are valid: $\vDash \{F\}\ S\ \{q\}$ and $\vDash \{p\}\ S\ \{T\}$. But neither triple says anything useful about the program S.
6. Under total correctness, we again have validity: $\vDash_{tot} \{F\}\ S\ \{q\}$, but again it says nothing useful about S. However, $\vDash_{tot} \{p\}\ S\ \{T\}$ says that if $\sigma \vDash p$, then running S in σ will terminate. We don't get any information about what the final state looks like, but at least there is one. As usual, if $\sigma \nvDash p$, then we know nothing about what will happen if you run S in σ.

7. False; $\sigma \vDash \{p\}\ S\ \{q\}$ does not imply $\sigma \vDash p$. (It doesn't imply $\sigma \nvDash p$ either.)
8. False; if $\sigma \in \Sigma$ and $\sigma \nvDash \{p\}\ S\ \{q\}$, then $\sigma \vDash p$ (and $M(S, \sigma) - \perp \vDash \neg q$).
9. True. $M(S, \sigma) \subseteq \{\perp_d, \perp_e\}$ says that S never terminates when run in σ. (It diverges or gets a runtime error.) Nontermination in σ implies partial correctness in σ.
10. True. $M(S, \sigma) \cap \{\perp_d, \perp_e\} \neq \varnothing$ says that S might not terminate when run in σ. I.e., at least one execution path causes an error: $\langle S, \sigma\rangle \rightarrow^* \langle E, \perp\rangle$. That path causes total correctness to fail: $\sigma \nvDash_{tot} \{p\}\ S\ \{q\}$.
11. True; if $\{p\}\ S\ \{q\}$ is partially correct and we run S in a state satisfying p, then either S causes an error or terminates in a state satisfying q.
12. False; if a triple is satisfied in σ but σ doesn't satisfy the precondition, then all possibilities can happen: S might diverge, it might cause a runtime error, and even if it terminates, the final state might satisfy q but it doesn't have to.
13. True; if partial correctness fails, it's because (for some execution path), running S terminates satisfying $\neg q$. If S is nondeterministic, it's still possible for there to be an execution path in which S terminates satisfying q.
14. True; we assumed $\perp \notin M(S, \sigma)$, so if total correctness fails, it's because running S can terminate satisfying $\neg q$. If S is nondeterministic, it's still possible for there to be an execution path in which S terminates satisfying q.

15. We're given $S \equiv x := x * x;\ y := y * y$ and $\sigma(x) = \alpha$ and $\sigma(y) = \beta$. For arbitrary σ,

$$\begin{aligned} M(S, \sigma) &= M(x := x * x;\ y := y * y,\ \sigma) \\ &= M(y := y * y,\ M(x := x * x,\ \sigma)) \\ &= M(y := y * y,\ \sigma[x \mapsto \alpha^2])) \\ &= \{\ \sigma[x \mapsto \alpha^2][y \mapsto \beta^2]\ \}. \end{aligned}$$

Since $\sigma(x) = \alpha$ and $\sigma(y) = \beta$., we know $\sigma \vDash x > y > 0$ implies $\alpha > \beta > 0$, which implies $\alpha^2 > \beta^2 > 0$, which implies $M(S, \sigma) = \{\ \sigma[x \mapsto \alpha^2][y \mapsto \beta^2]\ \} \vDash x > y > 0$.

This gives us total correctness right away, but just for practice we can analyze the situation for both partial correctness and termination. For partial correctness, we have $M(S, \sigma) - \bot = \{\ \sigma[x \mapsto \alpha^2][y \mapsto \beta^2]\ \} - \bot = \{\ \sigma[x \mapsto \alpha^2][y \mapsto \beta^2]\ \}$, which satisfies the postcondition, $x > y > 0$. For termination, $M(S, \sigma) - \bot = M(S, \sigma)$, so $\bot \notin M(S, \sigma)$, which means termination. Partial correctness plus termination gives us total correctness: $\sigma \vDash_{\text{tot}} \{x > y > 0\}\ S\ \{x > y > 0\}$.

[2022-09-26 - added]

16. The truly trivial cases are $\vDash \{p\}\ S\ \{q\}$ and $\vDash_{tot} \{p\}\ S\ \{q\}$ when
    - p is a contradiction
    - S always doesn't terminate
    - (For partial correctness) q is a tautology.
    - For total correctness, if q is a tautology, then the correctness triple says that S always terminates when started in a state that satisfies p. (We know the final state exists but we don't know anything else about it.)

[2022-09-26 - added]

17. $\vDash_{tot} \{p\}\ S\ \{q\}$ iff $\vDash \{p\}\ S\ \{q\}$ and $\vDash_{tot} \{p\}\ S\ \{T\}$.

