Proof Rules and Proofs

CS 536: Science of Programming, Fall 2022

Due Tue Oct 25, 11:59 pm

2022-10-20 pp. 1, 2; 2022-11-02 p.3, 2022-11-06 p.3

A. Why?

- To prove validity of correctness triples, we use a proof system with axioms for atomic statements and rules of inference for compound statements.

B. Outcomes

After this homework, you should be able to

- Verify and generate instances of the partial correctness proof rules.

C. Problems [60 points total]

Lectures 14 - 15: Proof Rules and Proofs, parts 1 & 2

For all the problems, if you define something using substitution notation (e.g., defining p' using "where $p' \equiv q'[e/v]$"), be sure to show the result of the substitution somewhere. Intermediate calculations that you write out might be worth partial credit.

Note the names used in one problem have no connection to the same names in other problems. (E.g., p_1 in Problem 1 is unrelated to p_1 in Problem 2.) Exception: Explicit connection can be made but they refer only to the given names. (E.g., if Problem 2 said "Let p_1 be as in Problem 1", then p_2 in Problems 1 and 2 are unrelated.)

You can use the looser sense of ≡ from lecture.

1. [12 = 4 * 3 points] Let $p \equiv x = 2\text{^}k \wedge k \le n$. Calculate p_1 and p_2, and the rule references R_1 and R_2.

   1. $\{p_1\}\ k := k+1\ \{p \equiv x = 2\text{^}k \wedge k \le n\}$ — *assignment (backward)*
   2. $\{p_2\}\ x := x*2\ \{p_1\}$ — *assignment (backward)*
   3. $\{p_2\}\ x := x*2;\ k := k+1\ \{p\}$ — *sequence 2, 1*
   4. $p \wedge k<n \rightarrow p_2$ — *predicate logic*
   5. $\{p \wedge k<n\}\ x := x*2;\ k := k+1\ \{p\}$ — R_1
   6. $\{\textbf{inv}\ p\}\ \textbf{while}\ k < n\ \textbf{do}\ x := x*2;\ k := k+1\ \textbf{od}\ \{p \wedge k \ge n\}$ [2022-10-20] — R_2

   Hint: $p_1 \equiv wp(k := k+1, p) \equiv \ldots$

2. [15 = 5 * 3 points] Let $p \equiv x = 2\text{^}k \wedge k \le n$. Calculate p_1, p_2, and p_3 and the rule references R_1 and R_2.

1. $\{p_1 \equiv p \wedge k < n\}\ x := x*2\ \{p_2\}$	*assignment (forward)*
2. $\{p_2\}\ k := k+1\ \{p_3\}$	*assignment (forward)*
3. $\{p_1\}\ x := x*2;\ k := k+1\ \{p_3\}$	*sequence 2, 1*
4. $p_3 \rightarrow p$	*predicate logic*
5. $\{p_1\}\ x := x*2;\ k := k+1\ \{p\}$	R_1
6. $\{$**inv** $p\}$ **while** $k < n$ **do** $x := x*2;\ k := k+1$ **od** $\{p \wedge k \ge n\}$ [2022-10-20]	R_2

Hints: $p_1 \equiv p \wedge k < n \equiv \ldots$?. $p_2 \equiv sp(p_1,\ x := x*2) \equiv \ldots$?

3. [33 = 11 * 3 points] Let
   - $q \equiv r = X*Y - x*y$
   - $IF \equiv$ ***if*** $even(x)$ ***then*** $y := 2*y;\ x := x/2$ ***else*** $r := r+y;\ x := x-1$ ***fi***
   - $even(x) \equiv x \,\%\, 2 = 0$, and $odd(x) \equiv x \,\%\, 2 \ne 0$

   Calculate $q_1 - q_6$ and $R_1 - R_5$, so that the proof of correctness below is correct. (For $R_1 - R_4$, say what kind of assignment is being used: *assignment (backward)* or *assignment (forward)*.)

1. $\{q_1\}\ x := x/2\ \{q\}$	$R_1 \equiv$ *assignment (???)*
2. $\{q_2\}\ y := 2*y\ \{q_1\}$	$R_2 \equiv$ *assignment (???)*
3. $q_3 \rightarrow q_2$	*predicate logic*
4. $\{q_3\}\ y := 2*y\ \{q_1\}$	*precondition strengthening 3, 2*
5. $\{q_3\}\ y := 2*y;\ x := x/2\ \{q\}$	*sequence 4, 1*
6. $\{q_4 \wedge r = r_0 \wedge x = x_0\}\ r := r+y\ \{q_5\}$	$R_3 \equiv$ *assignment (???)* *
7. $\{q_5\}\ x := x-1\ \{q_6\}$	$R_4 \equiv$ *assignment (???)*
8. $q_6 \rightarrow q$	*predicate logic*
9. $\{q_5\}\ x := x-1\ \{q\}$	*postcondition weakening 7, 8*
10. $\{q_4\}\ r := r+y;\ x := x-1\ \{q\}$	*sequence 6, 9*
11. $\{q\}\ IF\ \{q\}$	R_5

* We only use r_0 and x_0 in the false branch, and we drop them (in line 10) before forming the ***if-else*** (in line 11), so we don't have to add them to the true branch code or to the precondition of the ***if-else***.

Solution to Homework 7

1. $p_1 \equiv p[k+1/k] \equiv x = 2\text{^}(k+1) \wedge k+1 \le n$

   $p_2 \equiv p_1[x*2/x] \equiv x*2 = 2\text{^}(k+1) \wedge k+1 \le n$ [2022-11-02]

   $R_1 \equiv$ *precondition strengthening 4, 3*

   $R_2 \equiv$ *while loop, 5* †

2. $p_2 \equiv sp(p_1,\ x := x*2) \equiv (p \wedge k < n)[x_0 / x] \wedge x = x_0*2$

   $\equiv ((x = 2\text{^}k \wedge k \le n) \wedge k < n)[x_0 / x] \wedge x = x_0*2$ [fixes 2022-11-06]

   $\equiv (x_0 = 2\text{^}k \wedge k \le n \wedge k < n \wedge x = x_0*2)$

   $p_3 \equiv sp(p_2,\ k := k+1) \equiv p_2[k_0 / k] \wedge k = k_0+1$

   $\equiv (x_0 = 2\text{^}k \wedge k \le n \wedge k < n \wedge x = x_0*2)[k_0 / k] \wedge k = k_0+1$

   $\equiv (x_0 = 2\text{^}k_0 \wedge k_0 \le n \wedge k_0 < n \wedge x = x_0*2 \wedge k = k_0+1)$

   $R_1 \equiv$ *postcondition weakening 3, 4*

   $R_2 \equiv$ *while loop, 5*

[2022-11-06] Quick sanity check: $p_3 \rightarrow p$?

$p_3 \equiv (x_0 = 2\text{^}k_0 \wedge k_0 \le n \wedge k_0 < n \wedge x = x_0*2 \wedge k = k_0+1)$

$\Rightarrow x_0*2 = 2\text{^}(k_0+1) \wedge k_0+1 < n+1 \wedge (x = x_0*2 \wedge k = k_0+1)$

$\Rightarrow x = 2\text{^}k \wedge k < n+1$

$\Rightarrow x = 2\text{^}k \wedge k \le n$

$\equiv p$

3. $q \equiv r = X*Y - x*y$

   $q_1 \equiv wp(x := x/2,\ q) \equiv (r = X*Y - x*y)[x/2 / x] \equiv r = X*Y - (x/2)*y$

   $q_2 = wp(r := r+y,\ q_1) \equiv (r = X*Y - (x/2)*y)[2*y/y] \equiv r = X*Y - (x/2)*(2*y)$

   $q_3 \equiv q \wedge even(x) \equiv r = X*Y - x*y \wedge even(x)$

   $q_4 \equiv q \wedge odd(x) \equiv r = X*Y - x*y \wedge odd(x)$

   $q_5 \equiv sp(q_4,\ r := r+y) \equiv r_0 = X*Y - x*y \wedge odd(x) \wedge r = r_0+y$

   $q_6 \equiv sp(q_5,\ x := x-1) \equiv r_0 = X*Y - x_0*y \wedge odd(x_0) \wedge r = r_0+y \wedge x = x_0-1$

   $R_1, R_2 \equiv$ *assignment (backward)*

   $R_3, R_4 \equiv$ *assignment (forward)*

   $R_5 \equiv$ *conditional / if-else 5,10*

† We can be a little flexible with rule names: *while loop* and *loop* are ok; similarly *conditional* and *if-else* are ok.

