Syntactic Substitution, Forward Assignment, & sp

CS 536: Science of Programming, Fall 2022

Solution

For Problems 1 – 4, $p \equiv x-y<f(a) \vee \forall x.\, x \geq a*y \rightarrow \exists y.\, f(x-y)>a+y*z$.

1. $p[y+z/x]$
 $\equiv (x-y<f(a) \vee \forall x.\, x \geq a*y \rightarrow \exists y.\, f(x-y)>a+y*z)[y+z/x]$
 $\equiv (y+z)-y<f(a) \vee \forall x.\, x \geq a*y \rightarrow \exists y.\, f(x-y)>a+y*z$

2. $p[a-y/y]$
 $\equiv (x-y<f(a) \vee \forall x.\, x \geq a*y \rightarrow \exists y.\, f(x-y)>a+y*z)[a-y/y]$
 $\equiv x-(a-y)<f(a) \vee \forall x.\, x \geq a*(a-y) \rightarrow \exists y.\, f(x-y)>a+y*z$

3. $p[a*y/a]$
 $\equiv (x-y<f(a) \vee \forall x.\, x \geq a*y \rightarrow \exists y.\, f(x-y)>a+y*z)[a*y/a]$
 $\equiv x-y<f(a*y) \vee \forall x.\, x \geq (a*y)*y \rightarrow \exists y.\, f(x-y)>(a*y)+y*z$

4. $p[x \div y/a][y-z/x]$. First, let's do the initial substitution:
 $p[x \div y/a]$
 $\equiv (x-y<f(a) \vee \forall x.\, x \geq a*y \rightarrow \exists y.\, f(x-y)>a+y*z)[x \div y/a]$
 $\equiv x-y<f(x \div y) \vee \forall u.\, u \geq (x \div y)*y \rightarrow \exists v.\, f(u-v)>(x \div y)+v*z$
 (after renaming x to u and y to v)
 Then we can do the second substitution:
 $p[x \div y/a][y-z/x]$
 $\equiv (x-y<f(x \div y) \vee \forall u.\, u \geq (x \div y)*y \rightarrow \exists v.\, f(u-v)>(x \div y)+v*z)[y-z/x]$
 $\equiv (y-z)-y<f((y-z) \div y) \vee \forall u.\, u \geq ((y-z) \div y)*y \rightarrow \exists v.\, f(u-v)>((y-z) \div y)+v*z$

5. (Find an S such that $\models \{T\}\, S\, \{sp(T, S)\}$ but $\not\models_{\text{tot}} \{T\}\, S\, \{sp(T, S)\}$)
 For any σ, we know $M(S, \sigma)-\bot \models sp(T, S)$ for partial correctness. For lack of total correctness, we must have $\bot \in M(S, \sigma)$ to get $M(S, \sigma) \not\models sp(T, S)$. So we need an S that always fails to converge. Examples are $x := 1/0$ and ***while true do skip od***.

6. (Calculate an sp of two assignments)

   We want to calculate $sp(p, S_1; S_2) \equiv sp(sp(p, S_1), S_2)$ where $p \equiv x < y \wedge x + y \leq n$, $S_1 \equiv x := f(x+y)$, and $S_2 \equiv y := g(x*y)$. First,

   $sp(p, S_1) \equiv sp(x < y \wedge x + y \leq n, x := f(x+y)) \equiv x_0 < y \wedge x_0 + y \leq n \wedge x = f(x_0 + y)$

   Then $sp(sp(p, S_1), S_2)$ is $sp(x_0 < y \wedge x_0 + y \leq n \wedge x = f(x_0 + y), y := g(x*y))$

   $\equiv x_0 < y_0 \wedge x_0 + y_0 \leq n \wedge x = f(x_0 + y_0) \wedge y = g(x * y_0)$.

7. $sp(x < x*k, x := x/2) \equiv x_0 < x_0 * k \wedge x = x_0/2$.

   Simplifying, $x_0 < x_0 * k \Leftrightarrow (x_0/2) < (x_0/2)*k \Leftrightarrow x < x*k$.

8. $wp(x := x/2, x < x*k) \equiv x/2 < x/2*k$

9. (*sp* and *wp* of ***if-else***) We're given $S \equiv$ ***if*** $even(n)$ ***then*** $n := n-1$ ***fi***

   a. $wp(S, odd(n)) \equiv wp($***if*** $even(n)$ ***then*** $n := n-1$ ***else skip fi***$, odd(x))$

   $\equiv (even(n) \rightarrow wp(n := n-1, odd(n))) \wedge (odd(n) \rightarrow wp($***skip***$, odd(n)))$

   $\equiv (even(n) \rightarrow odd(n-1)) \wedge (odd(n) \rightarrow odd(n))$

   $\Leftrightarrow T$

   b. $sp(n = n_0, S) \equiv sp(n = n_0,$ ***if*** $even(n)$ ***then*** $n := n-1$ ***else skip fi***$)$

   $\equiv sp(n = n_0 \wedge even(n), n := n-1) \vee sp(n = n_0 \wedge odd(n),$ ***skip***$)$

   $\equiv even(n_0), n = n_0 - 1) \vee (n = n_0 \wedge odd(n))$

   $\Rightarrow odd(n_0) \wedge odd(n_0)$

   $\Leftrightarrow odd(n_0)$

