C241 Homework Assignment 9

1.

The language L and functions R, A, and T defined below are the same as in Section 7.6.

	$L \subseteq \{a,\ b,\ \bullet\}^+$
1.	$\bullet \in L$
2a.	$u \in L \Rightarrow au \in L$
2b.	$u \in L \Rightarrow bu \in L$
3.	*n. e.*

	$A\colon L^2 \to L$	$R\colon L \to L$	$T\colon L^2 \to L$
1.	$A(\bullet, v) = v$	$R(\bullet) = \bullet$	$T(\bullet, v) = v$
2a.	$A(au, v) = aA(u, v)$	$R(au) = A(R(u), a\bullet)$	$T(au, v) = T(u, av)$
2b.	$A(bu, v) = bA(u, v)$	$R(bu) = A(R(u), b\bullet)$	$T(bu, v) = T(u, bv)$

It is proved in the book that

- **Proposition 7.6.** *A is associative; that is, for all* $u, v, w \in L$, $A(u, A(v, w)) = A(A(u, v), w)$.
- **Proposition 7.9.** *Assuming Proposition 7.8, below, R is self-cancelling; that is, for all* $u \in L$, $R(R(u)) = u$.

Prove the following:

(a) **Proposition 7.7.** *For all* $u \in L$, $A(u, \bullet) = u$.

(b) **Proposition 7.8.** *For all* $u, v \in L$, $R(A(u, v) = A(R(v), R(u))$.

(c) **Proposition 7.11.** *For all* $u, v \in L$, $T(u, v) = A((R(u), v)$.

(d) **Proposition 7.10.** *For all* $u \in L$, $T(u, \bullet) = R(u)$.

(e) **Proposition 7.12.** *For all* $u \in L$, $T(T(u, \bullet), \bullet) = u$.

SOLUTION

(a) **Proposition 7.7.** *For all* $u \in L$, $A(u, \bullet) = u$.

PROOF. By induction on $u \in L$.

BASE CASE. $A(\bullet, \bullet) \overset{A.1}{=} \bullet$.

INDUCTION: Assume $IH \equiv A(u, \bullet) = u$.

$$A(au, \bullet) \overset{A.2a}{=} a\ A(u, \bullet) \overset{IH}{=} au$$

Similarly, for bu,

$$A(\text{b}u, \bullet) \overset{A.2b}{=} \text{b}\ A(u, \bullet) \overset{IH}{=} \text{b}u$$

(b) **Proposition 7.8.** *For all* $u, v \in L$, $R(A(u, v) = A(R(v), R(u))$. Proof. By induction on $u \in L$. Base Case.

$$\begin{array}{lcll} R(A(\bullet, v)) & = & R(v) & (A.1) \\ & = & A(R(v), \bullet) & (\text{Prop } 7.7) \\ & = & A(R(v), R(\bullet)) & (R.1) \end{array}$$

Induction.

(c) **Proposition 7.11.** *For all* $u, v \in L$, $T(u, v) = A((R(u), v)$.

Proof. By induction on $u \in L$.

Base Case.

$$\begin{array}{lcll} T(\bullet, v) & = & v & (T.1) \\ & = & A(\bullet, v) & (A.1) \\ & = & A(R(\bullet), v) & (R.1) \end{array}$$

Induction: Assume $T(u, v) = A(R(u), v)$.

$$\begin{array}{lcll} T(\text{a}u, v) & = & T(u, \text{a}v) & (T.2a) \\ & = & A(R(u), \text{a}v) & (\text{I.H.}) \\ & = & A(R(u), A(\text{a}\bullet, v)) & (A.2;\ A.1) \\ & = & A(A(R(u), \text{a}\bullet), v) & (\text{Prop. } 7.6) \\ & = & A(R(\text{a}u), v) & (R.2a) \end{array}$$

Similarly for $T(\text{b}u, v)$.

(d) **Proposition 7.10.** *For all* $u \in L$, $T(u, \bullet) = R(u)$.

Proof. Induction is not needed. By Proposition 7.11, $T(u, \bullet) = A(R(u), \bullet)$, and by Proposition 7.7, $A(R(u), \bullet) = R(u)$.

(e) **Proposition 7.12.** *For all* $u \in L$, $T(T(u, \bullet), \bullet) = u$.

Proof. Induction is not needed. By Proposition 7.10, used twice, $T(T(u, \bullet), \bullet) = T(R(u), \bullet) = R(R(u))$. And by Proposition 7.9, $R(R(u)) = u$.

2. The program below is called *Wensley's algorithm* for computing *the quotient of real numbers x and y to within tolerance t. Use the Theorem on Loop Invariants to prove the this program satisfies the post-condition* $\{z \leq x/y < z+t\}$.

```
{0 ≤ x < y ≤ 1}
begin
z := 0; d := 1; u := 0; v := ½y;
while d > t do
  { Inv ≡ z ≤ x/y < z+d ∧ u = zy ∧ v = ½dy}
  begin
  d := ½d;
  if u+v > x then skip
  else begin z := z+d; u := u+v; end ;
  v := ½v
  end
end
{z ≤ x/y < z+t}
```

SOLUTION

There are three things to prove:

INITIALIZATION. The program starts with $0 \leq x < y \leq 1$, and first reaches the while-loop with $z = 0$, $d = 1$, $u = 0$, and $v = \frac{1}{2}y$. The invariant INV is the conjunction of three propositions. All three are true when the variables' current values are substituted.

1. $z \leq x/y \leq z+d \longrightarrow 0 \leq x/y \leq 0+1$, which holds because $0 \leq x < y \leq 1$.
2. $u = zy \longrightarrow 0 = 0 \cdot y$, which is true.
3. $v = \frac{1}{2}dy \longrightarrow \frac{1}{2} \cdot 1 \cdot y = \frac{1}{2}y$, which is the initial value assigned to v.

INVARIANCE. If INV $\equiv z \leq x/y < z+d \;\wedge\; u = zy \;\wedge\; v = \frac{1}{2}dy$ holds and $d > t$ then the loop body executes, always computing new values for $d' = \frac{1}{2}d$ and $v' = \frac{1}{2}v$. The values of x and y are unchanged througout.

A. If conditional test $u + v > x$ is true, all other variables are unchanged.

1. $z' \leq x/y \leq z' + d' \longrightarrow z \leq x/y \leq z + \frac{1}{2}d$. The first part, $z \leq x/y$, is given by INV. Also by INV, $x < u + v = zy + \frac{1}{2}dy = (z + \frac{1}{2}d)y = (z' + d')y$. Dividing through by y preserves the inequality because y is positive and gives $x/y \leq z' + d'$ as desired.
2. $u' = z'y' \longrightarrow u = zy$, which is given by INV.
3. $v' = \frac{1}{2}d'y' \longrightarrow \frac{1}{2}v = \frac{1}{2}(\frac{1}{2}d)y$. Dividing this equation by $\frac{1}{2}$, we get $v = \frac{1}{2}dy$, which is given by INV.

B. If the conditional test, $u + v > x$ fails z and u are also changed and in addition to d' and v' we have.

z'	$=$	$z + d'$	because new assignment to d has already occurred
	$=$	$z + \frac{1}{2}d$	
u'	$=$	$u + v$	
	$\leq$	x	because the conditional test fails

Looking again at Inv one conjuct at a time,

1. $z' \leq x/y \leq z' + d' \longrightarrow z + \frac{1}{2}d \leq x/y \leq (z + \frac{1}{2}d) + \frac{1}{2}d$. $z + \frac{1}{2}d + \frac{1}{2}d = z + d$, and Inv gives us $x/y < z + d$, so the right-hand inequality is valid. To show $z + \frac{1}{2}d \leq x/y$, Inv gives us that $u = zy$ and $v = \frac{1}{2}dy$, so $u + v = (z + \frac{1}{2}d)y$. At this point of the program, $u + v \leq x$ and $y > 0$. Thus, $(z + \frac{1}{2}d)y \leq x$, and dividing both sides by y gives us what we want.
2. $u' = z'y' \longrightarrow u + v = (z + \frac{1}{2}d)y$. By Inv we have $u + v = zy + \frac{1}{2}dy = (z + \frac{1}{2}d)y$ as needed.
3. $v' = \frac{1}{2}d'y' \longrightarrow \frac{1}{2}v = \frac{1}{2}(\frac{1}{2}d)y$. Dividing this equation by $\frac{1}{2}$, we get $v = \frac{1}{2}dy$, which is given by Inv.

termination. On termination we have Inv $\wedge\ d \leq t$. So it is immediate that $z \leq x/y < z + d \leq z + t$, thus satisfying the postcondition.

3. Define $F\colon \mathbb{N} \to \mathbb{N}$ and $G\colon \mathbb{N}^2 \to \mathbb{N}$ as follows:

$$\begin{array}{rcl} F(0) & = & 1 \\ F(k+1) & = & (k+1) \times F(k) \end{array} \qquad \begin{array}{rcl} G(0,m) & = & m \\ G(k+1,m) & = & G(k,\ m \times (k+1)) \end{array}$$

(a) Prove by induction on $n \in \mathbb{N}$: For all $n, m \in \mathbb{N}$, $G(n,m) = m \times G(n,1)$.

(b) Prove: For all $n \in \mathbb{N}$, $F(n) = G(n,1)$.

SOLUTION

COMMENT. If you try to prove the result directly by induction, you'll get stuck:

Proposition 0. *For all $n \in \mathbb{N}$, $F(n) = G(n,1)$.*

BASE CASE.

$$F(0) = 1 = G(0,1)$$

INDUCTION. Assume that $F(k) = G(k,1)$.

$$\begin{array}{rcll} F(k+1) & = & (k+1) \times F(k) & \text{(Defn. } F) \\ & = & (k+1) \times G(k,1) & \text{(I.H.)} \\ & \vdots & ?! & (\textit{Nowhere to go from here\ldots} \\ & \stackrel{?}{=} & G(k+1,1) & \textit{\ldots to reach the goal.}) \end{array}$$

You need to find and prove a *more general* result about G.

Proposition 1. *For all $n, m \in \mathbb{N}$, $G(n,m) = m \times G(n,1)$.*

PROOF. The proof is by induction on $k \in \mathbb{N}$.

BASE CASE.

$$G(0,m) \stackrel{G}{=} m = m \times 1 \stackrel{F}{=} m \times F(0)$$

INDUCTION. Assume that $G(k,m) = m \times G(k,1)$. Then

$$\begin{array}{rcll} G(k+1,m) & = & G(k, m(k+1)) & \text{(Defn. } G) \\ & = & \big[m(k+1)\big] \times G(k,1) & \text{(I.H.} \\ & = & m \times \big[(k+1) \times G(k,1)\big] & \text{(algebra)} \\ & = & m \times G(k, k+1) & \text{(I.H.)} \\ & = & m \times G(k+1,1) & \text{(Defn. } G) \end{array}$$

□

COMMENT. A more "elegant" version of Proposition 1 would be *For all* $n, m, k \in \mathbb{N}$, $G(n, m \times k) = m \times G(n,k)$. However, we don't need that much generality for our purposes.

Corollary 2. *For all $n \in \mathbb{N}$, $F(n) = G(n, 1)$.*

PROOF. The proof is by induction on $k \in \mathbb{N}$.

BASE CASE.

$$F(0) = 1 = G(0, 1)$$

INDUCTION. Assume that $F(k) = G(k, 1)$.

$$\begin{array}{rcll} F(k+1) & = & (k+1) \times F(k) & \text{(Defn. } F\text{)} \\ & = & (k+1) \times G(k, 1) & \text{(I.H.)} \\ & = & G(k, k+1) & \text{(Prop. 1)} \\ & = & G(k+1, 1) & \text{(Defn. } G\text{)} \end{array}$$

□

4. Performance estimation for recursive programs often involves *recurrence relations* like the one below. Let $a \in \mathbb{N}$. The function $T\colon \mathbb{N} \to \mathbb{N}$ is defined recursively by

$$\begin{array}{rcl} T(0) & = & a \\ T(k+1) & = & T(k) + k + 1 \end{array}$$

We would like to find a *closed form* for T, that is, and an algebraic expression that does not involve recursion

Prove that for all $n \in \mathbb{N}$, $T(n) = a + \dfrac{n^2 + n}{2}$.

SOLUTION

PROOF. The proof is by induction on $n \in \mathbb{N}$.

BASE CASE.

$$T(0) = a = a + \frac{0 + 0^2}{2}$$

INDUCTION CASE. Assume that $T(k) = a + \dfrac{k^2 + k}{2}$.

$$\begin{array}{rcll} T(k+1) & = & T(k) + k + 1 & \text{(Defn. } T\text{)} \\ & = & a + \dfrac{k^2 + k}{2} + k + 1 & \text{(I.H.)} \\ & = & a + \dfrac{k^2 + k}{2} + \dfrac{2k + 2}{2} & \text{(multiply } k+1 \text{ by } 1 = \tfrac{2}{2}\text{)} \\ & = & a + \dfrac{k^2 + 3k + 2}{2} & \text{(adding fractions)} \\ & = & a + \dfrac{(k^2 + 2k + 1) + (k+1)}{2} & \text{(algebra)} \\ & = & a + \dfrac{(k+1)^2 + (k+1)}{2} & \text{(factoring } k^2 + 2k + 1\text{)} \end{array}$$

as needed. This completes the induction. □

Supplemental Problem. *H241 students should attempt this programming problem, but don't spend more than two or three hours on it.*

In a programming language of your choice, write a program that takes no input and outputs its own source code.

SOLUTION

No solution provided.