Assignment 1

Advanced Computing concepts for Engineering

Due Feb 8th, 2017 at 11:00am sharp.

Note that the work that you turn in for this assignment must represent your individual effort. You are welcome to help your fellow students to understand the material of the course and the meaning of the assignment questions, however, the answer that you submit must be created by you alone.

Q0 [10] Implementability and nondeterminism Again $\Sigma = \{\text{“}x\text{”} \mapsto \{0, 1, 2\}\}$
Consider the following specifications on $\Sigma \dagger \Sigma$

$$\begin{array}{rcl} f & = & \langle x' < x \rangle \\ g & = & \langle x' \leq x \rangle \\ h & = & \langle x = 2 - x' \rangle \\ \textbf{magic} & = & \langle \mathfrak{false} \rangle \\ \textbf{abort} & = & \langle \mathfrak{true} \rangle \end{array}$$

Which of these specifications are implementable? Which are nondeterministic? Explain why.

Q1 [15] Specification

Suppose $n > 0$ is a fixed member of $\mathbb{N}$. Remember that $\{0, ..n\}$ is the set containing the first n natural numbers. Then $\{0, ..n\} \overset{\text{tot}}{\to} \mathbb{R}$ is the set of all *sequences* of real numbers with a *length* of n. As an abbreviation, we'll write $\mathbb{R}^n$ for $\{0, ..n\} \overset{\text{tot}}{\to} \mathbb{R}$. We can use $\mathbb{R}^n$ as the type of real arrays of length n. Of course, if $a \in \mathbb{R}^n$ and $i \in \{0, ..n\}$, then $a(i)$ is item i of array a. The expression $a(i)$ is not defined if $a \in \mathbb{R}^n$ but $i \notin \{0, ..n\}$.

Let

$$\Sigma = \{\text{“}a\text{”} \mapsto \mathbb{R}^n, \text{“}b\text{”} \mapsto \mathbb{B}, \text{“}i\text{”} \mapsto \mathbb{N}\}$$

Write specifications on $\Sigma \dagger \Sigma$ for the problems below. The following function may be helpful

$$\begin{array}{rcl} \text{count} & \in & \mathbb{R}^n \times \mathbb{R} \overset{\text{tot}}{\to} \mathbb{N} \\ \text{count}(a, x) & = & |\{j \in \{0, ..n\} \mid a(j) = x\}| \end{array}$$

(a) [5] Reverse: The final value of a should be the reverse of its initial value.

(b) [5] Sorted: The final value of a is a nondecreasing sequence of values. I.e. each item should be greater or equal to all earlier items.

(c) [5] Permutation: The final value of a contains the same items as its initial value, in the same quantities, though perhaps not in the same order.

Q2 [10] Given $\Sigma = \{\text{“}x\text{”} \mapsto \mathbb{R}, \text{“}y\text{”} \mapsto \mathbb{R}\}$, implement the following specification using a sequence of (nonparallel) assignments.

$$\langle x' = y \wedge y' = x \rangle$$

Use forward substitution and erasure laws. (Hint use arithmetic operations.)

Q3 [10] Given $\Sigma = \{\text{“}x\text{”} \mapsto \mathbb{R}, \text{“}y\text{”} \mapsto \mathbb{R}, \text{“}z\text{”} \mapsto \mathbb{R}\}$, use the alternation law (and others) to implement

$$\langle (x < z \wedge y' = -1) \vee (x > z \wedge y' = 1) \vee (x = z \wedge y' = 0) \rangle$$

Q4 [20] Binary search. Given a constant $N > 0$ and a constant function $C : \{0, .., N\} \stackrel{\text{tot}}{\to} \mathbb{N}$, that is sorted (nondecreasing). We want to see whether there is an item of C that equals x. I'll use the notation $C\{p, ..r\}$ for the set of items with indecies in set $\{p, ..r\}$, i.e., $C\{p, ..r\} = \{k \in \{p, ..r\} \cdot C(k)\}$. Suppose we have variables p, q, r of type $\mathbb{N}$. Our specification is $f = \langle (x \in C\{0, ..N\}) = (x = C(p')) \rangle$. We can 'generalize' f as

$$g = \langle 0 \le p < r \le N \Rightarrow (x \in C\{p, ..r\}) = (x = C(p')) \rangle$$

(a) [5] Find an initialization command i such that $f \sqsubseteq i; g$.

(b) [10] Implement g recursively using an alternation. Try to ensure that each iteration reduces $r - p$ to roughly half its value. [Hint: Because C is nondecreasing: If $0 \le p < q < r \le N$ and $C(q) > x$, $(x \in C\{p, ..r\}) = (x \in C\{p, ..q\})$ .And if $C(q) \le x$ then $(x \in C\{p, ..r\}) = (x \in C\{q, ..r\})$.] Be sure to justify each step of your derivation.

(c) [5] Apply (the incomplete version of) the while law to implement g with a while-command. However you should still informally check that your loop will terminate, so state a bound expression for the loop.

Q5 [20] Russian peasant multiplication

Suppose x, y, and z are natural number variables. We want to implement: $f = \langle z' = x \times y \rangle$ without using a multiplication.

(a) [5] Find a 'generalization' g that will work with the initialization command $z := 0$. I.e. we want $f \sqsubseteq z := 0; g$.

(b) [10] Implement g recursively using an alternation. We would like the time to be proportional to the $\log_2 y$. You will find the following identities useful: $x \times y = x + x \times (y - 1)$ if $y > 0$. And $x \times y = 2 \times x \times y/2$ if y is even. Be sure to justify each step of your derivation.

(c) [5] Apply (the incomplete version of) the while law to implement g with a while-command. However you should still informally check that your loop will terminate. State a bound for your loop.