Assignment 0

Theodore S. Norvell

6892 Due 2017 Sept 28th

Q0 A linear search.

(a) [10] Using the verificaton rules in the notes, list all the conditions that must be shown to be universally true in order to show that the following proof outline is partially correct. Perform all substitutions. Assume state variables p and m hold integers and that a is an array of at least N integers

$\{A : N > 0\}$
$p, m := 1, a(0)$
$\{B : 1 \le p \le N \land (\forall i \in \{0, ..p\} \cdot a(m) \ge a(i))\}$
while $p < N$ do
    $\{C : 1 \le p < N \land (\forall i \in \{0, ..p\} \cdot a(m) \ge a(i))\}$
    if $a(p) \ge a(m)$ then
        $\{D : 1 \le p < N \land a(p) \ge a(m) \land (\forall i \in \{0, ..p\} \cdot a(m) \ge a(i))\}$
        $m := p$
    end if
    $\{E : 1 \le p < N \land (\forall i \in \{0, ..p + 1\} \cdot a(m) \ge a(i))\}$
    $p := p + 1$
end while
$\{F : \forall i \in \{0, ..N\} \cdot a(m) \ge a(i)\}$

(b) [5] Which of the conditions listed are universally true? Which are not?

Q1 [10] Log. All variables are integer.

Complete this proof outline so that it is correct and verifiable using the rules presented in the notes.

$\{n = N \ge 1\}$
...
$\left\{n \ge 1 \land n \times 2^k \le N < n \times 2^{k+1}\right\}$
while $n > 1$ do
    $\left\{n > 1 \land n \times 2^k \le N < n \times 2^{k+1}\right\}$
    ...
end while
$\left\{2^k \le N < 2^{k+1}\right\}$

Q2 [5] Window.

Suppose $a : \mathbb{N} \to \mathbb{R}$ is an infinite sequence. Define $f : \mathbb{Z}^+ \to \mathbb{R}$ by

$$f(i) = (a(i-1) + a(i) + a(i+1))/3, \text{ for all } i \in \mathbb{Z}^+$$

where $\mathbb{Z}^+ = \{i \in \mathbb{Z} \mid i > 0\}$.

Consider this algorithm in which x, y, and z are real, p and N are integer, b is real array of at least N items.

$\{N > 0\}$
$x, y := a(0), a(1)$
$p := 1$
$\{I\}$
while $p \neq N$ do
  $z := a(p+1)$
  $b(p) := (x + y + z)/3$
  $x, y := y, z$
  $p := p + 1$
end while
$\{\forall i \in \{1, ..N\} \cdot b(i) = f(i)\}$

State a loop invariant I that could be used to verify this algorithm.

Q3 [10] Polynomial evaluation (Design.)

Given a polynomial $p(x) = a(0)x^0 + a(1)x^1 + \cdots + a(k)x^k$, we can evaluate it without exponentiation by considering it as

$$p(x) = a(0) + x \times (a(1) + x \times (a(2) + x \times (\cdots a(k) \cdots)))$$

One way to put this a little more formally is that $p(x) = p_0(x)$, where

$$\begin{aligned} p_k(x) &= a(k) \\ p_i(x) &= a(i) + x \times p_{i+1}(x), \text{ for all } i \in \{0, ..k\} \end{aligned}$$

Write a correct and verifiable proof outline with precondition $\{true\}$ and postcondition $\{y = p(x)\}$. Assume k is an natural variable, x and y are real variables, and a is a sequence of at least $k+1$ real numbers. Your algorithm should not change x, a, or k and should not use exponentiation. Introduce any other variables as needed. Be sure to carefully state all loop invariants.

Bonus: Q2 used an assignment to an array, but no rule for assignments to arrays was given in class. Can you propose an verification rule for assignments to arrays?