HW 13 – Hoare Logic

CS 421 – Spring 2009

Revision 1.0

Assigned Friday, April 24, 2009
Due Wednesday, April 29, in class
Extension 48 hours (20% penalty)
Total points 50

1 Change Log

1.0 Initial Release.

2 Overview

You will practice on proofs in Hoare Logic.

3 Collaboration

Collaboration is allowed on this assignment.

4 Instructions

Write your answers in the blanks on the page. Submit in hard-copy at the beginning of class.

5 Problems (50 pts)

1. (10 pts) In the (annotated) notes for lecture 24, on page 10, a proof of the swap code is almost completely given. Complete it.

2. (10 pts) Similarly, on page 11, a proof of the absolute value code is almost completely given. Complete it

3. (10 pts) On page 14, the gcd algorithm is given. Give a proof tree. (Hint: the invariant of the loop is ”$\texttt{gcd}(\texttt{a}, \texttt{b}) = \texttt{gcd}(\texttt{a0}, \texttt{b0})$.”

4. (10 pts) Prove the judgment: $\texttt{true}\{\texttt{A}\}\texttt{m} = \texttt{max}(\texttt{a}[0], \texttt{a}[1], ..., \texttt{a}[\texttt{n} - 1])$, where A is this program:

```
m := a[0]; i := 1;}
while (i < n) {
   if (a[i] > m) m := a[i];
   i := i+1;
}
```

You may use any properties of the max function you like, such as $x > y \Rightarrow max(x, y) = x$, $x > y$ & $y = max(y, z) \Rightarrow x = max(x, y, z)$, and so on. (By the way, it was noted in class that the assignment rule does not work when subscripted arrays appear on the left-hand side of assignments, but that does not happen here, so you are free to use the assignment axiom where needed.)

5. (10 pts) following program calculates the number of positive elements in b:

```
c := 0;  a := b;
while (a != []) {
    if (hd a > 0)
    then c := c+1;
    a := tl a;
}
```

Give a loop invariant that could be used to prove the correctness of this program.