Homework 3 - Hoare Logic

March 16, 2012

Problem 1

We call a relation $r \subseteq S \times S$ functional if $\forall x, y, z \in S.(x, y) \in r \land (x, z) \in r \rightarrow y = z$. For each of the following statements either give a counterexample or prove it. In the following, assume $Q \subset S$.

(i) for any r, $wp(r, S \setminus Q) = S \setminus wp(r, Q)$

(ii) if r is functional, $wp(r, S \setminus Q) = S \setminus wp(r, Q)$

(iii) for any r, $wp(r, Q) = sp(Q, r^{-1})$

(iv) if r is functional, $wp(r, Q) = sp(Q, r^{-1})$

(v) for any r, $wp(r, Q_1 \cup Q_2) = wp(r, Q_1) \cup wp(r, Q_2)$

(vi) if r is functional, $wp(r, Q_1 \cup Q_2) = wp(r, Q_1) \cup wp(r, Q_2)$

(vii) for any r, $wp(r_1 \cup r_2, Q) = wp(r_1, Q) \cup wp(r_2, Q)$

(viii) Alice has the following conjecture: For all sets S and relations $r \subseteq S \times S$ it holds:

$$\Big(S \neq \emptyset \land dom(r) = S \land \triangle_S \cap r = \emptyset\Big) \rightarrow \Big(r \circ r \cap ((S \times S) \setminus r) \neq \emptyset\Big)$$

She tried many sets and relations and did not find any counterexample. Is her conjecture true? If so, prove it, otherwise provide a counterexample for which S is smallest.

Problem 2

Give a complete Hoare logic proof for the following program:

```
{n >= 0 && d > 0}
  q = 0
  r = n
  while ( r >= d ) {
    q = q + 1
    r = r - d
  }
{n == q * d + r && 0 <= r < d}
```

The proof should be step-by-step as in the example proof from the Hoare Logic Basics in the lecture notes (`http://lara.epfl.ch/w/sav09:hoare_logic_basics`). To prove each step you can use the syntactic rules for Hoare Logic (`http://lara.epfl.ch/w/sav09:syntactic_rules_for_hoare_logic`).

Problem 3

Consider the following code fragment:

```
while (x > 0) {
  y = y + x
  x = x - 3
}
```

Assume that the program state contains exactly the two variables, x and y, ranging over unbounded integers.

(i) Convert the program into guarded commands.

(ii) Find a formula $F(x, y, x', y')$ describing (precisely) the relationship between initial and final states of the code fragment. Show all steps you use to derive this formula. In particular, explicitly refer to each rule for constructing formulae that you use. Feel free to refer to relational semantics if needed (e.g. for transitive closure). The final result should be a closed-form formula that is as simple as possible. You can use all valid equations on integers and rules of first-order logic, but explain which rules you use. If some steps in finding the formula need proofs by induction, then carry out such proof. You may use the notation $k|x$ to denote that some integer constant k divides the value given by the expression x.

(iii) Let $r = \{((x, y), (x', y'))|F(x, y, x', y')\}$ be the relation computed for the program in the previous part. Let P be the formula $y == 0$. *Using the definition of strongest postcondition*, derive the strongest postcondition formula Q with respect to F, i.e. the formula Q such that the following is a Hoare triple

$$\{\ (x, y) \mid P\ \}\ \ r\ \ \{\ (x, y) \mid Q\ \}$$

Show your steps. You may use any math you need to express and manipulate the formulae, i.e. you are not restricted to Presburger, Peano or other arithmetic.

In the following, the precondition and postcondition formulae may use mathematical expressions if necessary, but should be as simple as possible. You may want to expand expressions like $k|x$ in order to be able to solve/simplify the formulae.

(iv) Using the definition of weakest precondition, derive the precondition formula P for the postcondition formula Q: $x < 0$. Show your steps.

(v) Using the definition of weakest precondition, derive the precondition formula P for the postcondition formula Q: $y < 0$. Show your steps.

(vi) Using the definition of weakest precondition, derive the precondition formula P for the postcondition formula Q: $x == y$. Show your steps.

(vii) Prove that $r \subseteq s$ where

$$s = \{((x, y), (x', y'))|(y == 0 \land x > 10) \rightarrow y' > 25\}$$

and $r = \{((x, y), (x', y'))|F(x, y, x', y')\}$ as before.