Quiz 2

Synthesis, Analysis, and Verification 2010

Friday, April 30, 2010

Last Name : ______________________________

First Name : ______________________________

Exercise	**Points**	**Achieved Points**
1	15	
2	15	
3	40	
4	40	
Total	110	

This quiz has many tasks. We therefore do not insist on writing your solution in too much detail, but please explain the key idea and give as much details as possible in the given time.

Problem 1: Fixpoints (15 points)

Let S be a set, $\mathit{Init}, \mathit{Good} \subseteq S$, and $r \subseteq S \times S$. Define the space of invariants by

$$\mathit{Invs} = \{I \subseteq S \mid \mathit{Init} \subseteq I \land sp(I, r) \subseteq I \land I \subseteq \mathit{Good}\}$$

Suppose $Invs \neq \emptyset$.
Define $h(X) = (\mathit{Init} \cup X) \cap \mathit{Good}$.
Define function f by

$$f(X) = h(X \cup sp(X, r))$$
$$g(X) = h(X \cap wp(X, r))$$

a) Is each element of *Invs* a fixpoint of f? Prove or give a counterexample.

b) Is each element of *Invs* a fixpoint of g? Prove or give a counterexample.

c)* If X is a fixpoint of both f and g, does it follow that it belongs to *Invs*? Prove or give a counterexample.

Problem 2: Predicate Abstraction (15 points)

Consider the set of predicates

$$\mathcal{P} = \{\text{false}, 0 \leq x, 0 \leq y, x \leq y\}$$

Let $A = 2^{\mathcal{P}}$. The meaning of a set of predicates $a \in A$, denoted $\gamma(a)$ is, as usual, the set of states that satisfies the conjunction of all predicates in a.
The precise semantics of a command `cmd` is the relation associated with `cmd`. For example, the precise semantics of the command

```
x = y; y = x + 1
```

in a program with two variables is the relation

$$\{((x, y), (x', y')) \mid x' = y \wedge y' = y + 1\}$$

For a given command `cmd` whose precise semantics is given by a relation r, let $sp^{\#}(a, \texttt{cmd})$ denote the least element $a' \in A$ such that $sp(\gamma(a), r) \subseteq \gamma(a')$.
As usual in programming languages, let `x++` denote a command that increments an integer variable `x` by 1 (assume that integer variables are unbounded).
Let $a_0 = \{0 \leq x, 0 \leq y, x \leq y\}$.
Compute the following values:

a) $\mathsf{sp}^{\#}(a_0, \texttt{x++})$

b) $\mathsf{sp}^{\#}(\mathsf{sp}^{\#}(a_0, \texttt{x++}), \texttt{y++})$

c) $\mathsf{sp}^{\#}(a_0, (\texttt{x++}; \texttt{y++}))$

Problem 3: Variable Range Analysis (40 points)

Consider the program in Figure 1.

```
x = 0;
y = 1;
while /* point H */
 (0 < x && x < 1000) {
  havoc(b);
  if (b > 0) {
    x := x + 1;
    y := 2 * x + 1
  } else {
    y := y - 1;
    x := x - 1
  }
}
/* point F */
```

Figure 1: Program with unbounded integer variables x,y,b, ranging over $\mathbb{Z}$.

a) Consider first a simple variable range analysis, which maintains an interval for each integer variable. Intervals can have as lower bound an integer or $-\infty$ and as the upper bound either an integer or $+\infty$. This analysis computes, for each statement, the smallest intervals that describe the values of variables after the statement. Your task is to indicate the intervals for each variable (x, y, b) that this analysis computes, both at the loop header `H` and the final program point `F`.

b) Suppose that we increase the loop bound in the program N times (from 1000 to $1000N$). Give an asymptotic function of N that indicates how the time to complete the analysis increases. (You can pick any iteration strategy for updating the values at different program points.)

c) In the rest of this problem, assume the loop bound is again 1000. Consider the analysis that uses widening with the set of jump points $J = \{-\infty, -1, 0, 1, 1000, +\infty\}$, by applying at each step the widening function w (which maps the interval into a smallest enclosing interval whose bounds are in J). Assuming that the analysis has reached the fixpoint, indicate the intervals for x, y, b at the loop header `H` and the final program point `F`.

d) Indicate the values of variables at `H` and `F` when we apply narrowing iteration until the fixpoint (that is, we apply the computation as in a), without widening, but starting from the result computed in part c)).

Problem 4: Analysis of Relations between Variables (40 points)

In this problem we explore how to automatically derive $\mathsf{sp}^{\#}(\_)$ when our analysis domain is given by a system of integer linear inequalities.
Consider a program with two integer variables, x, y. Let P be the set of states satisfying

$$x - 2y \le a_1 \ \wedge \ x - y \le a_2$$

where a_1 and a_2 are some parameters.
Consider a relation r mapping states (x, y) into states (x', y') given by

$$x' = 3y + 1 \ \wedge y' = x + y$$

a) Write down a quantified formula $Q(x', y')$ such that

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

for every a_1, a_2.

b)* Write down a quantifier-free formula $Q_0(x', y')$ equivalent to $Q(x', y')$. You can use any mathematically correct reasoning steps to derive formula $Q_0(x', y')$; there is no need to restrict yourself to a specific algorithm.

c)* Suppose that we wish to find the strongest approximation of $Q_0(x', y')$ that has the same syntactic form as P. That is, we search for formula $Q_1(x', y', b_1, b_2)$ of the form

$$x' - 2y' \le b_1 \ \wedge \ x' - y' \le b_2 \qquad (*)$$

for some b_1, b_2. We wish to find the values b_1 and b_2 as a function of a_1, a_2 such that $Q_0(x', y')$ implies $Q_1(x', y', b_1, b_2)$ and such that for any other values b_1', b_2' for which $Q_0(x', y')$ implies $Q_1(x', y', b_1', b_2')$, we have that the formula $Q_1(x', y', b_1', b_2')$ is implied by $Q_1(x', y', b_1, b_2)$.

Describe as efficient as possible procedure to generate expressions for functions that map a_1, a_2 into b_1, b_2.

Explain how to generalize your procedure to arbitrary preconditions, and arbitrary relations r expressed in integer linear arithmetic.