EPFL Formal Verification Course, Quiz 2019-12-12

Each subquestion is scored independently.

Wrong answers do not penalize other parts.

You are allowed to use every true statement from the lecture slides provided that you refer to the lecture in which it is stated.

1 Invariants (16 points)

```
import stainless.collection._

object EvenCount {
  def evenCountAcc[T](l: List[T], p: T ⇒ Boolean, acc: Boolean): Boolean = {
    l match {
      case Nil() ⇒ acc
      case Cons(x, xs) ⇒ evenCountAcc(xs, p, p(x) != acc)
    }
  }

  def evenCount[T](l: List[T], p: T ⇒ Boolean): Boolean = {
    evenCountAcc(l, p, true)
  }
}
```

Figure 1: EvenCount

Consider the function `evenCount` given in Figure 1.

Question 1: Explain what `evenCount` function computes
(1 point)

Question 2: Write a postcondition for the value `res` returned by `evenCount`. Your postcondition $P($`res`, `l.count(p)`$)$ must be expressed in terms of `res` and `l.count(p)` (the latter counts the number of elements `x` in `l` such that `p(x)` holds). You can use standard arithmetic and boolean operations.
(5 points)

Question 3: Write a postcondition for the value returned by the auxiliary function `evenCountAcc` to prove that `evenCount` computes what you expressed in questions 1 and 2. Your postcondition $Q($`res`, `acc`, `l.count(p)`$)$ must be expressed in terms of `res`, `acc`, and `l.count(p)`.
(7 points)

Question 4: Explain why:

1. If the postcondition Q holds for the call `evenCountAcc(l, p, true)`, in `evenCount`, then the postcondition P holds for `evenCount`, i.e. why

   $$Q(\texttt{res},\ \texttt{true},\ \texttt{l.count(p)}) \Rightarrow P(\texttt{res},\ \texttt{l.count(p)})$$

   holds (with all variables being universally quantified).
2. The postcondition Q holds in the `Nil` branch of `evenCountAcc`, i.e. why

   $$Q(\texttt{acc},\ \texttt{acc},\ \texttt{Nil().count(p)})$$

   holds (with all variables being universally quantified).
3. The postcondition Q holds in the `Cons` branch of `evenCountAcc`, assuming the value returned by the recursive call `evenCountAcc(xs, p, p(x)` $\neq$ `acc)` respects the postcondition). Said otherwise, explain why:

   $$Q(\texttt{res},\ \texttt{p(x)} \neq \texttt{acc},\ \texttt{xs.count(p)}) \Rightarrow Q(\texttt{res},\ \texttt{acc},\ \texttt{Cons(x,xs).count(p)})$$

   holds (again with all variables being universally quantified).

(3 points)

2 Iterating sp and wp (28 points)

This question is inspired by relationships in a transition system. Let S be a non-empty set of states and $r \subseteq S \times S$. As usual, define

- for every $P \subseteq S$, $sp(P, r) = \{x' \mid \exists x \in P.(x, x') \in r\}$
- for every $Q \subseteq S$, $wp(r, Q) = \{x \mid \forall x'.(x, x') \in r \to x' \in Q\}$

Let $C = 2^S = \{X \mid X \subseteq S\}$ be the set of all sets of states. Let also: $Init \subseteq S$ and $Good \subseteq S$. Define:

- $F : C \to C$ as $F(X) = Init \cup sp(X, r)$
- $B : C \to C$ as $B(Y) = Good \cap wp(r, Y)$
- $l = \bigcup_{i \geq 0} F^i(\emptyset)$
- $h = \bigcap_{i \geq 0} B^i(S)$

The ordering relation we consider is $\subseteq$ relation on C. For each of the following determine if the property holds or not. If it holds, prove it (If you do not know how to prove it or you run out of time, explain why you believe it to be true for partial points). If you believe the property to be false, give a counterexample (no need to prove that it is a counterexample).

1. F is monotonic
2. B is monotonic
3. l is the least fixed point of F
4. h is the greatest fixed point of B
5. $l \subseteq h$
6. $l \subseteq Good$ implies $Init \subseteq h$
7. $Init \subseteq h$ implies $l \subseteq Good$

Each part carries 4 points.

3 Fixed Points in Lattices (23 points)

Consider a complete lattice with the set of elements L and the partial order $\sqsubseteq$. (Remember that in a complete lattice, every set $X \subseteq L$ has the least upper bound and the greatest lower bound.)

Let $G : L \to L$ be a monotonic function with respect to $\sqsubseteq$. Let

$$Fix = \{x \in L \mid G(x) = x\}$$

be the set of all fixed points. Let $x, y \in Fix$ be two fixed points. Prove or disprove:

1. $x \sqcup y \sqsubseteq G(x \sqcup y)$ (4 points)
2. $G(x \sqcup y) \sqsubseteq x \sqcup y$ (6 points)
3. $x \sqcup y \in Fix$ (4 points)
4. Let $B = \{b \in Fix \mid x \sqsubseteq b \wedge y \sqsubseteq b\}$. Then B has the least element, that is, an element $z \in B$ such that $\forall b \in B.\ z \sqsubseteq b$ (9 points. Possibly difficult.)

4 Reasoning about Simple Integer Linear Inequalities (24 points)

Fix a set of variables V and consider as the set $\mathcal{F}$ of conjunctions of literals of the form $x \le y + c$ where $x, y \in V$ and where c is an arbitrary integer constant (possibly different for each literal):

$$\bigwedge_{i=1}^{k} x_{p_i} \le x_{q_i} + c_i$$

For example, if $V = \{x, y, z\}$, an example $F \in \mathcal{F}$ is:

$$x \le y + 1 \land y \le z + (-2) \land z \le x + 0$$

1. give an algorithm that, for $F \in \mathcal{F}$ and a variable $y \in V$, computes a formula $G \in \mathcal{F}$ that is equivalent to $\exists y.F$
   (8 points)

2. describe an algorithm that for $F \in \mathcal{F}$ determines if F is satisfiable, i.e., there exist values of variables for which formula is true
   (8 points)

3. describe an algorithm that for $F, G \in \mathcal{F}$ checks whether $F \to G$ is a valid formula
   (8 points)

Analyze the efficiency of your algorithms and make them polynomial if possible.

5 Composing Galois Connections (9 points)

Galois connection (named after Évariste Galois) is defined by two monotonic functions $\alpha : C \to A$ and $\gamma : A \to C$ between partial orders $\leq$ on C and $\sqsubseteq$ on A, such that

$$\forall c, a. \quad \alpha(c) \leq_A a \iff c \leq_C \gamma(a) \qquad (*)$$

Denote the fact that α, γ are monotonic and that $(*)$ holds by

$$C \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} A$$

Assume we have two Galois connections:

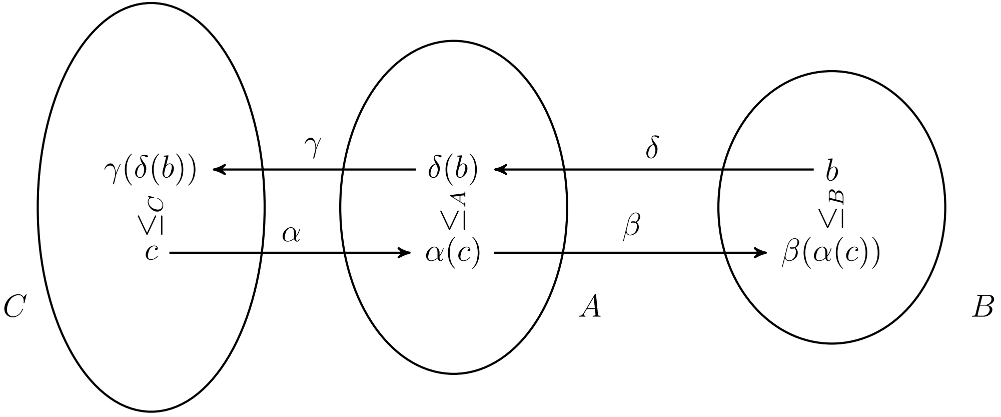
$$C \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} A \quad \text{and} \quad A \underset{\beta}{\overset{\delta}{\leftrightarrows}} B$$

Prove that $\beta \circ \alpha$ and $\gamma \circ \delta$ form a Galois connection between C and B:

$$C \underset{\beta\circ\alpha}{\overset{\gamma\circ\delta}{\leftrightarrows}} B$$

For any two functions, we define composition by $(f \circ g)(x) = f(g(x))$.

Here is a larger picture of the situation:



Each of the three sets C, A, B has its own ordering, but feel free to denote them all simply by $\leq$ if there is no danger that you confuse yourself.