

# Program Analysis

## ASSIGNMENT 2

Due on Thursday Nov 19, 2009

---

1. Remember the definition of liveness; a property  $S$  is a liveness property if and only if:

$$Lim \circ PCI(S) = \Sigma^\infty \iff PCI(S) = \sigma^+ \iff \Sigma^+ \subseteq PCI(S)$$

Show that proving *liveness* properties imperatively requires the consideration of infinite behaviours, that is:

$$\forall \alpha \in \Sigma^+ \exists \beta \in \Sigma^\infty : \alpha\beta \in S$$

2. Express in LTL the following properties (assuming trace  $\sigma = s_0s_1\dots$ ):

- If  $p$  occurs at least twice, then  $p$  occurs infinitely often.
- $p$  holds at all states  $s_{3k}$  ( $0 \leq k$ ) and does not hold at all states  $s_{3k+1}$  and  $s_{3k+2}$ .
- if  $p$  holds at a state  $s_i$ , then  $q$  holds at at least one of the two states just before  $s_i$ , that is  $s_{i-1}$  and  $s_{i-2}$ .
- $p$  never holds in less than two consecutive states; that is, if  $p$  holds at a state  $s_i$ , then it also hold at  $s_{i-1}$  or  $s_{i+1}$ .

3. Generalize the following *functional abstraction* to a *relation abstraction* by switching function  $@$  with a relation:

- $D$  and  $D'$  are sets
- $\langle L, \sqsubseteq, \top, \perp, \sqcup, \sqcap \rangle$  is a complete lattice.
- $@ : D \mapsto D'$
- $\alpha \in (D \mapsto L) \mapsto (D' \mapsto L) = \lambda y. \sqcup \{f(x) \mid @(x) = y\}$
- $\gamma \in (D' \mapsto L) \mapsto (D \mapsto L) = g \circ @$  and prove that your  $\alpha$  and  $\gamma$  make a Galois connection.

4. We add a new loop construct to SIL:

```
for (X from n to m) do
...
od
```

It is your typical *for* loop. The increment is always 1, and the range is determined by the lower bound  $n$  and the upper bound  $m$ . When  $n \leq m$ , the loop body executes exactly  $m - n + 1$  times, and when  $n > m$ , it does not execute at all.  $X$  is assigned value  $n$  in the first iteration,  $n + 1$  in the second iteration, and so on. If there are other decisions to be made about the meaning of this new loop construct, you are free to make your own choices (as long as it makes sense!). Provide the small step operational semantics to the *for* loop in the style of lecture (6).