CS615 Homework #2

Max marks: 60 **Due Oct 3, 2007**

- *Be brief, complete and stick to what has been asked.*
- ***Do not copy solutions from others.***

1. We have discussed in class how to reason about programs using the abstract domain of Difference Bound Matrices (DBM). For a program with n integer variables $x_1, \ldots x_n$, the elements of the abstract domain lattice are $(n+1) \times (n+1)$ matrices, where the 0^{th} row and column correspond to the special variable x_0 that always has the value 0. For purposes of this question, we will assume that the element in the i^{th} row and j^{th} column gives an integer upper bound of $x_i - x_j$, i.e., $x_i - x_j \leq M[i][j]$.

   Consider the following program in our simple programming language:

```
L1:  x1 := 10;
L2:  x2 := 20;

L4:  while (f(x1, x2, x3)) {
L5:     if (g1(x1, x2, x3) && (x1 > 20) {
L6:        x1 := 0;
L7:     }
L8:    else if ((x1 <= 20) && (x2 <= 30)) {
L9:       x1 := x1 + 2;
L10:       x2 := x2 + 1;
L11:    }
L12: }
```

   In the above program, `f(x1, x2, x3)` and `g1(x1, x2, x3)` are boolean valued functions. No further information is provided about these functions – this is a common way of abstracting complex functions simply using their return types.

   (a) *[10 marks]* Compute as good a loop invariant as you can at `L4` using the abstract domain of DBMs. In computing your loop invariant, use the *lub* operation for the first two iterations, and use the widen operation subsequently. Clearly show each step of your calculation.

   (b) Now suppose our abstract domain is enriched, so that each element in the abstract domain lattice is a set of DBMs, with each set having at least one and at most two DBMs. Thus, an abstract domain element is either a singleton set of one DBM or a set of two DBMs. Given such an abstract domain element, the concrete set of states it represents is the union of the concrete sets of states represented by each DBM in the set. The partial ordering relation between elements in the new abstract domain is one that is induced by the subset partial ordering in the concrete domain.

      i. *[5+5+5 marks]*Give pseudo-code of an algorithm to compute *lub*, *glb* and widen operators in this new abstract domain, by using the *lub*, *glb* and widen operators on DBMs (the base abstract domain). You may assume that you have at your disposal a function `check(set1, set2)` to check the partial ordering between two sets of DBMs. This function returns `true` if `set1` is lower in the partial order compared to `set2`, and `false` otherwise.

ii. *[15 marks]* Compute as good a loop invariant as you can at `L4` of the above program using this new abstract domain. You must follow the same idea of using *lub* for the first two iterations, using widen subsequently.

2. Let $(\mathcal{P}(S), \subseteq, \emptyset, S, \cup, \cap)$ and $(\mathcal{L}, \sqsubseteq, \bot, \top, \sqcup, \sqcap)$ be complete lattices, where $\mathcal{P}(S)$ denotes the powerset of S. Let $\alpha : \mathcal{P}(S) \to \mathcal{L}$ and $\gamma : \mathcal{L} \to \mathcal{P}(S)$ be a Galois connection pair.

   Prove the following:

   (a) *[10 marks]* $\alpha(b_1 cup b_2) \;=\; \alpha(b_1) \sqcup \alpha(b_2)$, for all $b_1, b_2 \in \mathcal{P}(S)$.

   (b) *[10 marks]* $\gamma(a_1 \sqcap a_2) \;=\; \gamma(a_1) \cap \gamma(a_2)$ for all $a_1, a_2 \in \mathcal{L}$.