Program Synthesis meets Machine Learning

Assignment 1 — Due on: 29.01.2020

1. **Hoare Logic**.

   (a) Indicate whether or not the following Hoare triple is valid, based on the rules discussed in class. Justify your answer. (Small credit - S)

   $\{$ *true* $\}$
   `while` $(x \,!= 0)$
   $\quad x = x + 1;$
   $\{\ x = 0\ \}$

   (b) Consider the following *non-deterministic assignment* construct: (Medium credit - M)

   $\quad x := [v_1, v_2];$

   Here variable x is non-deterministically assigned the value of v_1 or v_2.

   i. Consider the Hoare triple

   $$\{\ \mathrm{P}\ \}\ \ x := [v_1, v_2]\ \ \{\ \phi(x)\ \},$$

   where ϕ is some predicate containing x. What is the weakest precondition P?

   ii. Consider the Hoare triple

   $$\{\ \phi(x)\ \}\ \ x := [v_1, v_2]\ \ \{\ \mathrm{Q}\ \},$$

   where ϕ is some predicate containing x. what is the strongest postcondition Q?

   (c) For the following program, write the loop invariant. Also prove the validity of the Hoare triple, using the rules discussed in class. (M)

   $\{\ n \geq 1\ \}$
   sum = 0;
   $i = 1;$
   `while` $(i \leq n)$
   `do`
   $\quad$ *temp* = $i * i$
   $\quad$ *sum* = *sum* + *temp*;
   $\quad i = i + 1;$
   $\{\ (i > n) \wedge\ ($*sum* $= n * (n + 1) * (2n + 1)/6)\ \}$

2. **Propositional Logic**. Prove the validity/satisfiability of the following formulas: (S)

   (a) $(p \Rightarrow p) \Rightarrow p$

   (b) $(p\ \wedge q\ \Rightarrow\ r)\ \wedge\ (p\ \wedge \neg q\ \Rightarrow\ r)\ \Leftrightarrow\ (p\ \Rightarrow\ r)$

   (c) $p\ \wedge (\neg p \vee q)\ \wedge\ (\neg q\ \vee\ r)\ \wedge\ (\neg r\ \vee\ \neg p)$

3. **Davis Putnam Logeman Loveland (DPLL)**. Prove the validity/satisfiability of the following formula using DPLL method: (M)

   $(p \ \vee \ (\neg q \ \wedge \ r)) \Leftrightarrow ((q \ \vee \neg r) \ \Rightarrow \ p)$

   (a) Convert the formula into NNF form.

   (b) Convert the resulting formula into CNF form.

   (c) Use DPLL method to prove your claim on the CNF formula.

4. **EUF**. Apply the congruence closure algorithm to decide the satisfiability of the following EUF formula. (M)

   $(x = z) \ \wedge \ (\ f(g(x,y)) = g(x, f(y)) \ ) \ \wedge \ (\ g(f(x), f(y)) = y \ ) \ \wedge \ (f(z) \neq x)$

5. **PL/FOL**.

   A finite graph is a pair $G = (V, E)$ where V is a finite set of vertices and E is a finite set of 2-element subsets of V called the edge set. Given a set of $k-$colors $C = \{c_1, c_2, \ldots c_k\}$, the *2-hop coloring* for G is to assign a color $c \in C$ for each vertex $v \in V$ such that for every vertex in $H_2(v)$ has a different color, where $H_2(v)$ contains all vertices which are at most 2-hops away from v (i.e, vertex $u \in H_2(v)$ iff $u \in V$ and G has a path between v and u that has at most two edges).

   Show how to encode an instance of a *2-hop coloring* problem into a propositional formula F that is satisfiable iff a *2-hop coloring* exists. (M)

   (a) Describe a set of propositional constraints asserting that every vertex is colored. Use the notation $color(v) = c_i$ to indicate that a vertex v has the color c_i. Such an assertion is encodable as a single propositional variable $p_v^{c_i}$.

   (b) Describe a set of propositional constraints asserting that every vertex has at most one color.

   (c) Describe a set of propositional constraints asserting that for every vertex $v \in V$ no two vertices in $H_2(v)$ have the same color.

   (d) Specify your constraints in CNF.

6. **Nelson-Oppen**. Apply Nelson-Oppen procedure to check satisfiability of the following formula: (M)

$$F = \begin{array}{ll} & f(f(z) + f(x)) \neq f(f(y) + f(x)) \\ \wedge & x \leq y + z \\ \wedge & z \geq 0 \\ \wedge & w \geq 0 \\ \wedge & x \leq y \end{array}$$