COMSM0067: Advanced Topics in Programming Languages
Problem Sheet: Hoare Logic versus Symbolic Execution

What does each one tell you?

---

1. Consider

$$\{n \geq 0\} \;\; P_1 \;\; \{s = n \cdot n\}$$

where P_1 is

$$i := 0; \quad s := 0; \quad \mathsf{while}\ i < n\ \mathsf{do}\ \big(s := s + 2 \cdot i + 1;\ i := i + 1\big).$$

   (a) Prove the triple using Hoare logic, with the invariant

$$I \;\equiv\; s = i \cdot i \;\wedge\; 0 \leq i \leq n.$$

   (b) Execute P_1 symbolically, with n the symbolic input α and the precondition as the initial path condition, unrolling the loop at most **three** times. For each exiting path, give the path condition and $\sigma(s)$, and decide whether the postcondition can fail. If it can, give a concrete input.

   (c) What did you prove in (a), and what in (b)?

2. (*) Consider

$$\{n \geq 0\} \;\; P_2 \;\; \{s = n(n-1)/2\}$$

where P_2 is

$$i := 0; \quad s := 0; \quad \mathsf{while}\ i \leq n\ \mathsf{do}\ \big(s := s + i;\ i := i + 1\big).$$

   (a) Prove the triple using Hoare logic, with the invariant

$$I \;\equiv\; s = i(i-1)/2 \;\wedge\; 0 \leq i \leq n + 1.$$

   (b) Execute P_2 symbolically, as in 1(b), unrolling the loop at most **three** times. For each exiting path, give the path condition and $\sigma(s)$, and decide whether the postcondition can fail. If it can, give a concrete input.

   (c) What did you prove in (a), and what in (b)?

Reference

Weakest preconditions. $\{P\}\ c\ \{Q\}$ holds iff $P \Rightarrow \mathsf{wp}(c, Q)$.

$$\begin{aligned} \mathsf{wp}(x := a,\ Q) &= Q[a/x] \\ \mathsf{wp}(c_1;\ c_2,\ Q) &= \mathsf{wp}(c_1,\ \mathsf{wp}(c_2,\ Q)) \end{aligned}$$

Loops. There is no wp rule for while: we supply a loop invariant I instead. To prove

$$\{P\}\ \ c_0;\ \mathsf{while}\ b\ \mathsf{do}\ c\ \ \{Q\}$$

it suffices to show the three verification conditions

$$\begin{array}{lll} \text{VC1} & \text{(init)} & P \Rightarrow \mathsf{wp}(c_0,\ I) \\ \text{VC2} & \text{(preserve)} & I \wedge b \Rightarrow \mathsf{wp}(c,\ I) \\ \text{VC3} & \text{(exit)} & I \wedge \neg b \Rightarrow Q \end{array}$$

Symbolic execution. A symbolic state is a pair (σ, π): a *symbolic store* σ, mapping each variable to an expression over the symbolic inputs, and a *path condition* π, what the inputs must satisfy to reach this point.

$$\begin{array}{ll} x := a & (\sigma, \pi) \longrightarrow (\sigma[x \mapsto \sigma(a)],\ \pi) \\ \text{guard } b & (\sigma, \pi) \text{ forks into } (\sigma,\ \pi \wedge \sigma(b)) \text{ and } (\sigma,\ \pi \wedge \neg\sigma(b)) \end{array}$$

Here $\sigma(a)$ is a with each variable replaced by its value in σ. A path whose π is unsatisfiable is *infeasible* and is dropped. A while loop is *unrolled*: its guard forks once per iteration, up to the bound.

Can the postcondition Q fail on a path? Ask whether

$$\pi \wedge \neg\sigma(Q)$$

is satisfiable. If so, a model is a concrete input on which Q fails.