In Class Exercises – Axiomatic Semantics

1. Classify the following Hoare triples as valid or invalid, and explain why briefly. Examples:

   - $\{\textbf{true}\}\ x := x * x\ \{x \geq 0\}$ VALID; x^2 is always ≥ 0.
   - $\{x \geq 0\}\ x := x - 1\ \{x \geq 0\}$ INVALID; If $x = 0$ at the start of the program, then $x \geq 0$ but $x - 1 < 0$.

   (a) $\{\textbf{true}\}\ x := 3\ \{x = 4\}$

   (b) $\{\textbf{true}\}\ x := 3\ \{\textbf{false}\}$

   (c) $\{\textbf{false}\}\ x := 3\ \{\textbf{false}\}$

   (d) $\{x > y \wedge x > 0\}\ y := x + y\ \{y > 0\}$

   (e) $\{y > x \wedge x > 0\}\ x := x + y\ \{x > y\}$

   (f) $\{x = 3\}$ **while** $odd(x)$ **do skip** $\{\textbf{true}\}$

   (g) $\{x\ is\ even\}$ **while** $odd(x)$ **do skip** $\{\textbf{true}\}$

   (h) $\{\textbf{false}\}$ **while** $odd(x)$ **do skip** $\{x\ is\ even\}$

2. Use the assignment rule to find a precondition Q that makes each triple valid.

   (a) $\{Q\}\ x := x * 3\ \{x = 15\}$

   (b) $\{Q\}\ x := 14\ \{x = 14\}$

   (c) $\{Q\}\ r := r - 1\ \{r^3 + sin(r) - 12 = 4\}$

   (d) $\{Q\}\ y := y + 1; x := x + a[y]\ \{x = \sum_{k=1}^{y} a[k]\}$

3. Fill in the blanks with the appropriate assertions to come up with Q. Use the pairs of blank lines to show uses of the consequence rule (i.e., to show simplifications).

   Q : __________

   ____

   if $w = z$ **then begin**

       ____

       ____

       $w := 4;$

       ____

       ____

       $z := z \div 2$

       ____

     end

   else begin

       ____

       ____

       $z := w * z;$

       ____

       ____

       $w := 2 * w$

       ____

     end

   $\{z = 5 * w\}$

4. To prove the loop

   ```
   {pre Q}
   Initialization
   {inv P}
   while B do S;
   {post R}
   ```

   correct, we use the following steps:

   - Prove $\{Q\}$ Initialization $\{P\}$ is valid
   - Prove P is invariant: $\{P \wedge B\}\ S\ \{P\}$
   - Prove that if the loop stops, R holds: $P \wedge \neg B \Rightarrow R$
   - Prove bound function t decreases with each iteration.
   - Prove that if there are iterations left then $t > 0$: $P \wedge B \Rightarrow t > 0$

   Prove the following loop correct:

   ```
   {0 < n}
   i := 1;
   {inv P:   (0 < i ≤ n) ∧ (i is a power of 2)}
   {bound function:   n − i}
   while 2 ∗ i ≤ n do i := i ∗ 2;
   {(0 < i ≤ n < 2 ∗ i) ∧ (i is a power of 2)}
   ```

5. We can also go the other way: Given a precondition, invariant, and postcondition, we can develop a corresponding loop using the following procedure:

   - Find a loop guard B. ("When are we done?")
   - Find initialization to establish invariant P.
   - Guess bound function t and find ways to decrease it. (Make sure it meets the requirements listed in the first section.)
   - Ensure that P is reestablished.

   Write a program that, given n, calculates

   $$F_n = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F_{n-1} + F_{n-2} & \text{if } n > 1 \end{cases}$$

   Given the following specification:

   ```
   {pre Q:   n > 0}
   {inv P:   1 ≤ i ≤ n ∧ a = F_i ∧ b = F_{i−1}}
   {post R:   a = F_n}
   ```