Sample Solutions to Homework #2

1. (Section 3.6, Exercise 2, page 290) We use IF to denote the whole if statement. To prove \( \{ \text{true} \} \text{ IF } \{ x \geq 0 \} \), we need to prove \( \text{true} \Rightarrow wp(IF, x \geq 0) \). We first use the \( wp \) method to compute:

\[
\begin{align*}
wp(IF, x \geq 0) & \equiv \{ wp \text{ for if statements} \} \\
& \equiv (x < 0 \land wp(x := 0, x \geq 0)) \lor (x \geq 0 \land wp(\text{skip}, x \geq 0)) \\
& \equiv \{ wp \text{ for assignments and skip} \} \\
& \equiv (x < 0 \land 0 \geq 0) \lor (x \geq 0 \land x \geq 0) \\
& \equiv \{ \text{arithmetic; logic} \} \\
& \equiv x < 0 \lor x \geq 0 \\
& \equiv \{ \text{logic} \} \\
& \text{true},
\end{align*}
\]

and clearly, \( \text{true} \Rightarrow \text{true} \). The correctness of the code with respect to the given pre- and post-conditions is thus established.

2. (Section 3.6, Exercise 6, page 290) We again use IF to denote the whole if statement. To prove \( \{ \text{true} \} \text{ IF } \{ y = 2 \} \), we need to prove \( \text{true} \Rightarrow wp(IF, y = 2) \). We first compute:

\[
\begin{align*}
wp(IF, x \geq 0) & \equiv \{ wp \text{ for if statements; simplification of conditions} \} \\
& \equiv (x < 0 \land wp(y := -2|x|/x, y = 2)) \lor (x > 0 \land wp(y := 2|x|/x, y = 2)) \lor (x = 0 \land wp(y := 2, y = 2)) \\
& \equiv \{ wp \text{ for assignment statements} \} \\
& \equiv (x < 0 \land -2|x|/x = 2) \lor (x > 0 \land 2|x|/x = 2) \lor (x = 0 \land 2 = 2) \\
& \equiv \{ \text{arithmetic and logic; since } x < 0 \Rightarrow -2|x|/x = 2, \text{ hence } x < 0 \land -2|x|/x = 2 \equiv x < 0 \} \\
& \equiv x < 0 \land x > 0 \lor x = 0 \\
& \equiv \{ \text{logic} \} \\
& \text{true},
\end{align*}
\]

and clearly, \( \text{true} \Rightarrow \text{true} \). The correctness of the code with respect to the given pre- and post-conditions is thus established.

3. (Section 3.6, Exercise 12, page 290) We choose (after a few trials and failures) \( a = dq + r \land r \geq 0 \) to be our loop invariant \( I \). (Note that there may be other loop invariants.) To prove the correctness of the code, we need to prove three claims.
1. \{a > 0 \land b > 0\} \ r := a; \ q := 0 \ \{I\}

To prove this claim, we first compute:

\[
wp(r := a; \ q := 0, I) \\
\equiv \wp \text{ for sequencing} \\
wp(r := a, wp(q := 0, I)) \\
\equiv \wp \text{ for assignments; simplification} \\
wp(r := a, a = r \land r \geq 0) \\
\equiv \wp \text{ for assignments} \\
a = a \land a \geq 0 \\
\equiv \text{arithmetic; logic} \\
a \geq 0
\]

We then need to prove: \(a > 0 \land b > 0 \Rightarrow a \geq 0\), which is clearly true by arithmetic.

2. \{I \land r \geq d\} \ r := r - d; \ q := q + 1 \ \{I\}

To prove this claim, we first compute:

\[
wp(r := r - d; \ q := q + 1, I) \\
\equiv \wp \text{ for sequencing} \\
wp(r := r - d, wp(q := q + 1, I)) \\
\equiv \wp \text{ for assignments} \\
wp(r := r - d, a = d(q + 1) + r \land r \geq 0) \\
\equiv \wp \text{ for assignments} \\
a = d(q + 1) + (r - d) \land r \geq 0 \\
\equiv \text{simplification} \\
a = dq + r \land r \geq 0,
\]

which is just \(I\). We next need to prove: \(I \land r \geq d \Rightarrow I\), which is again clearly true by weakening in logic.

3. \{I \land r < d\} \Rightarrow \{a = dq + r \land 0 \leq r < d\}

It is straightforward to see that

\[
I \land r < d \\
\equiv \{I \equiv a = dq + r \land r \geq 0\} \\
a = dq + r \land 0 \leq r < d,
\]

which is the same as the right hand side, thus proving the claim.

4. (a) By the definition of the \(M\) function, \(M(n) = n - 10\), for all \(n \geq 101\). We use strong induction to prove that \(M(n) = 91\), for all \(n \leq 101\). (Note that when \(n = 101\), \(n - 10 = 91\), so the two definitions, i.e., \(M(n) = n - 10\) and \(M(n) = 91\), do not conflict.) Inducting on \(n\) involves a technique called “reverse induction,” where the value of \(n\) decreases, instead of increases. To
avoid this complexity, we let $n = 101 - k$ and we do strong induction on $k$ (so that when $k$
increases, $n$ decreases). We first prove the base case, which involves 11 (not just one) values
of $k$. Consider $k = 0, 1, \ldots, 10$, in that order. When $k = 0$, $M(101 - k) = M(101) = 91$.
When $k = 1$, $M(101 - k) = M(100) = M(M(111)) = M(101) = 91$. Using a similar method,
we can verify that $M(101 - k) = 91$ for all $k = 0, 1, \ldots, 10$. We next carry out the inductive
step. Let $k > 10$ and assume that $M(101 - \ell) = 91$ for all $0 \leq \ell < k$. We need to prove that
$M(101 - k) = 91$. We observe that
\[
M(101 - k) \\
= \{\text{since } k > 10, \text{ hence } 101 - k \leq 100; \text{ by the definition of } M\} \\
M(M(101 - k + 11)) \\
= \{\text{arithmetic}\} \\
M(M(101 - (k - 11))) \\
= \{\text{treat } k - 11 \text{ as } \ell, \text{ and observe that } 0 \leq \ell < k; \text{ inductive hypothesis}\} \\
M(91) \\
= \{\text{base case}\} \\
91.
\]
This completes the inductive step and the whole proof.

(b) The iterative implementation is given in class, repeated here.
\[
\begin{align*}
i &:= 1; \\
\textbf{while} \ (i > 0) \ {\{} \\
\text{if} \ (n > 100) \ {\{} n := n - 10; \ i := i - 1 \ {\}} \\
\text{else} \ {\{} n := n + 11; \ i := i + 1 \ {\}} \\
{\}}
\end{align*}
\]

(c) The key part is to find an appropriate loop invariant. Consider $M(N) = M^{(i)}(n) \land i \geq 0$
as the loop invariant $I$ (there may be other possible loop invariants), where $M^{(i)}(n)$ means
applying $M$ to $n$ for $i$ times. We use IF to denote the if statement inside the loop. We then
need to prove three claims:

1. $\{n = N\} i := 1 \ {I}\$
   To prove this claim, we need to prove $n = N \Rightarrow \text{wp}(i := 1, I)$. We first compute:
   \[
   \text{wp}(i := 1, I) \\
   \equiv \{I \equiv M(N) = M^{(i)}(n) \land i \geq 0\} \\
   \text{wp}(i := 1, M(N) = M^{(i)}(n) \land i \geq 0) \\
   \equiv \{\text{wp for assignments}\} \\
   M(N) = M(n) \land 1 \geq 0 \\
   \equiv \{\text{arithmetic}\} \\
   M(N) = M(n).
   \]
   Thus, we need to prove $n = N \Rightarrow M(N) = M(n)$, which is immediately true.
2. \( \{ I \land i > 0 \} \) IF \( \{ I \} \)

To prove this claim, we need to prove \( I \land i > 0 \Rightarrow wp(\text{IF}, I) \). We first compute:

\[
wp(\text{IF}, I) \\
\equiv \{ wp \text{ for } IF \} \\
\quad (n > 100 \land wp(n := n - 10; \; i := i - 1, I)) \lor \\
\quad (n \leq 100 \land wp(n := n + 11; \; i := i + 1, I)) \\
\equiv \{ wp \text{ for sequencing} \} \\
\quad (n > 100 \land wp(n := n - 10, wp(i := i - 1, I))) \lor \\
\quad (n \leq 100 \land wp(n := n + 11, wp(i := i + 1, I))) \\
\equiv \{ wp \text{ for assignments} \} \\
\quad (n > 100 \land M(N) = M^{(i-1)}(n) \land i - 1 \geq 0) \lor \\
\quad (n \leq 100 \land M(N) = M^{(i+1)}(n) \land i + 1 \geq 0) \\
\quad \text{(strengthening, because } i \geq 1 \Rightarrow i \geq -1) \\
\quad (n > 100 \land M(N) = M^{(i-1)}(n - 10) \land i \geq 1) \lor \\
\quad (n \leq 100 \land M(N) = M^{(i+1)}(n + 11) \land i \geq 1) \\
\quad \text{(definition of } M; \; n > 100 \Rightarrow n - 10 = M(n); \; n \leq 100 \Rightarrow M(M(n + 11)) = M(n)) \\
\quad (n > 100 \land M(N) = M^{(i)}(n) \land i \geq 1) \lor \\
\quad (n \leq 100 \land M(N) = M^{(i)}(n) \land i \geq 1) \\
\equiv \{ \text{simplification; logic} \} \\
\quad M(N) = M^{(i)}(n) \land i \geq 1.
\]

We then need to prove \( I \land i > 0 \Rightarrow M(N) = M^{(i)}(n) \land i \geq 1 \), which is immediately true. Hence, \( I \land i > 0 \Rightarrow wp(\text{IF}, I) \).

3. \( I \land i \leq 0 \Rightarrow n = M(N) \)

This claim is true, because

\[
I \land i \leq 0 \\
\equiv \{ I \equiv M(N) = M^{(i)}(n) \land i \geq 0 \} \\
\quad M(N) = M^{(i)}(n) \land i \geq 0 \land i \leq 0 \\
\equiv \{ \text{arithmetic} \} \\
\quad M(N) = M^{(i)}(n) \land i = 0 \\
\equiv \{ \text{logic} \} \\
\quad M(N) = M^{(0)}(n) = n.
\]

(d) One possible potential function (there may be others) is: \( \Delta = 2|n - 111| + 21i \). To see this indeed is a potential function, first note that \( \Delta \) is always non-negative. Next, if IF takes the first branch, then \( 2|n - 111| \) increases by at most 20 because \( |n - 111| \) increases by at most 10, while \( 21i \) decreases by exactly 21 because \( i \) decreases by exactly 1. Hence, \( \Delta \) decreases by at least 1. If IF takes the second branch, then \( 2|n - 111| \) decreases by exactly 22 because \( |n - 111| \) decreases by exactly 11, and \( 21i \) increases by exactly 21 because \( i \) increases by exactly 1. Thus, \( \Delta \) decreases by exactly 1. Hence, \( \Delta \) decreases at each iteration and is always non-negative. Thus, by the potential function argument, the loop terminates eventually.