

Exercise Sheet 2

General remarks:

- **Due date:** December 16th (before the exercise class).
- Solutions must be written in English.
- While we *might* publish sketches of exercise solutions, we do *not* guarantee that these sketches contain all details that are necessary to properly solve an exercise. Hence, it is recommended to attend the exercise classes.
- If you have any questions regarding the lecture or the exercise, feel free to write us an email or visit us at the chair.

Exercise 1 (Properties of Weakest Preconditions)

35%

Let c be a GCL program and f, g be predicates.

- [10%] Show that $\text{wp}[c](0) = 0$.
- [15%] Show that $f \Rightarrow g$ implies $\text{wp}[c](f) \Rightarrow \text{wp}[c](g)$.
- [10%] Does $\text{wp}[c](f) \vee \text{wp}[c](g)$ hold iff $\text{wp}[c](f \vee g)$? If yes, why? If not, find special cases for which this law holds.

Solution: (a) By structural induction on c .

The case `skip`: $\text{wp}[\text{skip}](0) = 0$

The case `abort`: $\text{wp}[\text{abort}](0) = 0$

The case $x := E$: $\text{wp}[x := E](0) = 0[x \mapsto E] = 0$

The case `if`(G){ c_1 }`else`{ c_2 }:

$$\begin{aligned}
 & \text{wp}[\text{if}(G)\{c_1\}\text{else}\{c_2\}](0) \\
 &= (G \Rightarrow \text{wp}[c_1](0)) \wedge (\neg G \Rightarrow \text{wp}[c_2](0)) \\
 &= (G \Rightarrow 0) \wedge (\neg G \Rightarrow 0) && \text{(I.H. twice)} \\
 &= G \wedge \neg G \\
 &= 0
 \end{aligned}$$

The case $c_1; c_2$

$$\begin{aligned}
 & \text{wp}[c_1; c_2](0) \\
 &= \text{wp}[c_1](\text{wp}[c_2](0)) \\
 &= \text{wp}[c_1](0) && \text{(I.H.)} \\
 &= 0 && \text{(I.H.)}
 \end{aligned}$$

The case $\{c_1\} \sqcap \{c_2\}$:

$$\begin{aligned}
 & \text{wp}[\{c_1\} \sqcap \{c_2\}](0) \\
 &= \text{wp}[c_1](0) \wedge \text{wp}[c_2](0) \\
 &= 0 \wedge 0 && \text{(I.H. twice)} \\
 &= 0
 \end{aligned}$$

The case while (G) {c}: Let $F_0(X) = (G \Rightarrow \text{wp}[c](X)) \wedge (\neg G \Rightarrow 0) = \text{wp}[c](X)$. By definition, we have $\text{wp}[\text{while } (G) \{c\}](0) = \sup_{n \in \mathbb{N}} F_0^n(0)$. Since $F_0(0) = \text{wp}[c](0) = 0$ by I.H., it follows that $\text{wp}[\text{while } (G) \{c\}](0) = 0$.

Solution: (b) Let $f \Rightarrow g$. By structural induction on c .

The case skip: $\text{wp}[\text{skip}](f) = f \Rightarrow g = \text{wp}[\text{skip}](g)$

The case abort: $\text{wp}[\text{abort}](f) = 0 = \text{wp}[\text{abort}](g)$. Hence, $\text{wp}[\text{abort}](f) \Rightarrow \text{wp}[\text{abort}](g)$.

The case $x := E$: $\text{wp}[x := E](f) = f[x \mapsto E] \Rightarrow g[x \mapsto E] = \text{wp}[x := E](g)$ (note that this holds due to the pointwise ordering of predicates)

The case ite (G) {c₁} {c₂}

$$\begin{aligned} & \text{wp}[\text{ite } (G) \{c_1\} \{c_2\}](f) \\ &= (G \Rightarrow \text{wp}[c_1](f)) \wedge (\neg G \Rightarrow \text{wp}[c_2](f)) \\ &\Rightarrow (G \Rightarrow \text{wp}[c_1](g)) \wedge (\neg G \Rightarrow \text{wp}[c_2](g)) && \text{(I.H. twice)} \\ &= \text{wp}[\text{ite } (G) \{c_1\} \{c_2\}](g) \end{aligned}$$

The case $c_1; c_2$

$$\begin{aligned} & \text{wp}[c_1; c_2](f) \\ &= \text{wp}[c_1](\text{wp}[c_2](f)) \\ &\Rightarrow \text{wp}[c_1](\text{wp}[c_2](g)) && \text{(I.H.)} \\ &= \text{wp}[c_1; c_2](g) \end{aligned}$$

The case $\{c_1\} \sqcap \{c_2\}$:

$$\begin{aligned} & \text{wp}[\{c_1\} \sqcap \{c_2\}](f) \\ &= \text{wp}[c_1](f) \wedge \text{wp}[c_2](f) \\ &\Rightarrow \text{wp}[c_1](g) \wedge \text{wp}[c_2](g) && \text{(I.H. twice)} \\ &= \text{wp}[\{c_1\} \sqcap \{c_2\}](g) \end{aligned}$$

The case while (G) {c}. For every predicate h , let $F_h(X) = (G \Rightarrow \text{wp}[c](X)) \wedge (\neg G \Rightarrow h) = \text{wp}[c](X) \vee (\neg G \wedge h)$. By definition, (and task 3) we know that $\text{wp}[\text{while } (G) \{c\}](f) = \sup_n F_f^n(0)$.

We first show for all $n \geq 0$ that $F_f^n(0) \Rightarrow F_g^n(0)$ by complete induction on n . Then it follows immediately (by the monotone sequence theorem) that

$$\text{wp}[\text{while } (G) \{c\}](f) = \sup_n F_f^n(0) \Rightarrow \sup_n F_g^n(0) = \text{wp}[\text{while } (G) \{c\}](g)$$

The remaining induction proceeds as follows:

$$I.B. \ n = 0: F_f^0(0) = 0 = F_g^0(0)$$

I.S. $n \mapsto n + 1$:

$$\begin{aligned}
F_f^{n+1}(0) &= (G \Rightarrow \text{wp}[c](F_f^n(0)) \wedge \neg G \Rightarrow f) \\
&\Rightarrow (G \Rightarrow \text{wp}[c](F_g^n(0)) \wedge \neg G \Rightarrow f) && \text{(I.H.)} \\
&\Rightarrow (G \Rightarrow \text{wp}[c](F_g^n(0)) \wedge \neg G \Rightarrow g) && (f \Rightarrow g) \\
&= F_g^{n+1}(0)
\end{aligned}$$

Solution: (c)

One implication does *not* hold in the presence of nondeterministic choice. For example, consider the program c given by $\{x := 0\} \square \{x := 1\}$. Then

$$\text{wp}[c](x = 0 \vee x = 1) = (0 = 0 \vee 0 = 1) \wedge (1 = 0 \vee 1 = 1) = 1.$$

However,

$$\text{wp}[c](x = 0) \vee \text{wp}[c](x = 1) = (0 = 0 \wedge 1 = 0) \vee (1 = 0 \wedge 1 = 1) = 0.$$

In the absence of nondeterministic choice, however, **wp** is distributive with respect to disjunction. This can again be shown by structural induction on c as in the previous two tasks.

Exercise 2 (Calculation of Weakest Pre-conditions)

20%

Consider the program c given by

$$x := X ; z := 0 ; \text{while } (y \leq x) \{ z := z + 1 ; x := x - y \}.$$

with respect to postcondition $z = X \text{ div } y$, where div denotes integer division.

- (a) [10%] Provide an invariant I for the loop of c w.r.t. postcondition $z = X \text{ div } y$.
- (b) [10%] Prove that your proposed invariant is correct.

Solution: A possible invariant is

$$I = y \geq 0 \wedge x \geq 0 \wedge z + (x \text{ div } y) = X \text{ div } y.$$

Let *body* denote the loop body of c . We now show that I is in fact an invariant, i.e., $G \wedge I \Rightarrow \text{wp}[\text{body}](I)$.

$$\begin{aligned}
&G \wedge I \Rightarrow \text{wp}[\text{body}](I) \\
&\text{iff } G \wedge I \Rightarrow y \geq 0 \wedge x - y \geq 0 \wedge z + 1 + (x - y \text{ div } y) = X \text{ div } y \\
&\text{implies } G \wedge I \Rightarrow y \geq 0 \wedge x \geq 0 \wedge z + 1 + (x \text{ div } y) - 1 = X \text{ div } y && (0 \leq y \leq x) \\
&\text{implies } G \wedge I \Rightarrow y \geq 0 \wedge x \geq 0 \wedge z + (x \text{ div } y) = X \text{ div } y \\
&\text{iff } G \wedge I \Rightarrow I,
\end{aligned}$$

where the last statement is equivalent to true.

Exercise 3 (Alternative Semantic Characterization for Loops)**24%**

Recall that

$$\text{wp}[\text{while } (G) \{c\}](f) = \mu F = \sup_n F^n(\mathbf{0}) ,$$

where F is the expectation transformer $X \mapsto [G] \cdot \text{wp}[c](X) + [\neg G] \cdot f$ and F^n denotes the composition of F with itself n times, i.e. $F^0 = id$ and $F^{n+1} = F \circ F^n$.

Prove by induction on n that

$$\forall n. F^n(\mathbf{0}) = \text{wp}[\text{while}_n(G) \{c\}](f) ,$$

where $\text{while}_n(G) \{c\}$ represents the n^{th} -unrolling of the loop given by

$$\begin{aligned} \text{while}_0(G) \{c\} &= \text{abort} \\ \text{while}_{n+1}(G) \{c\} &= \text{ite}(G) \{c; \text{while}_n(G) \{c\}\} \{\text{skip}\} . \end{aligned}$$

Solution:

Base case:

$$\begin{aligned} &\text{wp}[\text{while}_0(G) \{c\}](f) \\ &= \{\text{def. while}_0(G) \{c\}\} \\ &\text{wp}[\text{abort}](f) \\ &= \{\text{def. wp}\} \\ &\mathbf{0} \\ &= \{\text{def. } F^0\} \\ &F^0(\mathbf{0}) \end{aligned}$$

Inductive case:

$$\begin{aligned} &\text{wp}[\text{while}_{n+1}(G) \{c\}](f) \\ &= \{\text{def. while}_{n+1}(G) \{c\}\} \\ &\text{wp}[\text{ite}(G) \{c; \text{while}_n(G) \{c\}\} \{\text{skip}\}](f) \\ &= \{\text{def. wp multiple times}\} \\ &[G] \cdot \text{wp}[c](\text{wp}[\text{while}_n(G) \{c\}](f)) + [\neg G] \cdot f \\ &= \{\text{I.H.}\} \\ &[G] \cdot \text{wp}[c](F^n(\mathbf{0})) + [\neg G] \cdot f \\ &= \{\text{def. } F\} \\ &F(F^n(\mathbf{0})) \\ &= \{\text{def. } F^{n+1}\} \\ &F^{n+1}(\mathbf{0}) \end{aligned}$$

Exercise 4 (Program Specifications)**21%**

UPDATE 13.12.2016: Since the weakest liberal precondition (wlp) calculus has not been introduced in the lecture yet, you can ignore part (a), (d), and (e) of task 4.

Match each of the following formal specifications of program c

- (a) $[3\%] \ [P] = \text{wlp}[c](1)$
- (b) $[3\%] \ [P] = \text{wp}[c](1)$
- (c) $[3\%] \ [P] \leq \text{wp}[c](1)$
- (d) $[3\%] \ 1 = \text{wlp}[c](0)$
- (e) $[3\%] \ 1 \leq \text{wlp}[c](0)$
- (f) $[3\%] \ 1 = \text{wp}[c](0)$
- (g) $[3\%] \ 0 \leq \text{wp}[c](Q)$

with their corresponding colloquial description from 1–7. (There may be more than one formal specification in (a)-(g) with the same colloquial description in 1–7. There may also be “unmatched” colloquial interpretations.)

1. Program c diverges almost surely for all initial states.
2. The program never finishes in a final state satisfying Q .
3. The specification does not say anything about program c ; it is logically equivalent to **true**.
4. The specification is logically equivalent to **false**.
5. None of the above. Provide yourself the interpretation of the specification.
6. Program c terminates almost surely whenever executed in an initial state that satisfies P .
7. Program c terminates almost surely when executed in an initial state that satisfies P and diverges almost surely when executed in an initial state that satisfies $\neg P$.

Solution: (a)-3 (b)-7 (c)-6 (d)-1 (e)-1 (f)-4 (g)-3
--