Exercise Sheet 6

General remarks:

- **Due date:** December 9th 12:30 (before the exercise class). Note that this sheet is worth 150 points but you have two weeks to work on it.
- Please submit your solutions via MOODLE. Remember to provide your matriculation number. It is necessary to hand in your solutions in groups of **three**. You may use the MOODLE forum to form groups.
- Solutions must be written in English.
- While we will 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, please use the forum in MOODLE.

Exercise 1 (Probabilistic invariants) — 30P

Consider the following pGCL program P:

$$\texttt{while}\ (x > 0)\ \{\ \{\ x := x - 1\ \}\ [p]\ \{\ x := x + 1\ \}\ \}\ .$$

For $p = 1/3$ you have already seen in the lecture that $I = [x > 0] \cdot 2^{-x} + [x \leq 0]$ is a wp-superinvariant[1] of P wrt. the constant post-expectation $f = 1$.

(a) [2P] Give an non-trivial[2] upper bound on the probability that the program terminates when started with $x = 3$. Assume that $p = 1/3$.

(b) [25P] Now let $p \in [0, 1]$. Find a *wp*-superinvariant $I_p \neq 1$ (depending on p) of P wrt. to post-expectation $f = 1$ and prove that your answer is correct.

(c) [3P] Use your result from (b) to compute a non-trivial upper bound on the probability that the program terminates when started in $x = 100$, assuming $p = 0.49$.

Solution:

(a)

Since I is a wp-superinvariant of P wrt. to post-expectation 1, we know that $wp[\![P]\!](1) \sqsubseteq I$. However, for a given initial state s, $wp[\![P]\!](1)(s)$ is just the probability to terminate starting in s. Therefore we simply have to evaluate $I = [x > 0] \cdot 2^{-x} + [x \leq 0]$ at initial state $s = \{x \mapsto 3\}$. This yields

$$[3 > 0] \cdot 2^{-3} + [3 \leq 0] \ = \ 2^{-3} \ = \ 0.125\ .$$

Thus the probability to terminate assuming that $x = 3$ initially is at most 0.125.

(b)

We try the invariant *template* $I_p = [x > 0]y^{-x} + [x \leq 0]$. That is, we *guess* that there is a suitable y (depending on p) such that I_p becomes a wp-superinvariant. Now we try to find such

[1] In fact, I is the *least* fixed point of Φ_f.

[2] A *non-trivial* upper bound on a probability is a bound that is strictly smaller than 1.

a y. We do this by considering the inequality $\Phi_1(I_p) \sqsubseteq I_p$ to find suitable sufficient conditions for y. The wp-characteristic function Φ_1 is as follows:

$$\Phi_1(X) = [x > 0]\,(pX[x/x-1] + (1-p)X[x/x+1]) + [x \leq 0]$$

$$\begin{aligned}
\Phi_1(I_p) &= [x>0]\left(p([x-1>0]y^{-(x-1)} + [x-1\leq 0]) + (1-p)([x+1>0]y^{-(x+1)} + [x+1\leq 0])\right) + [x\leq 0] \\
&= [x>0]\left(p([x>1]y^{-x+1} + [x\leq 1]) + (1-p)([x>-1]y^{-x-1} + [x\leq -1])\right) + [x\leq 0] \\
&= p[x>0][x>1]y^{-x+1} + p[x>0][x\leq 1] + (1-p)[x>0][x>-1]y^{-x-1} + (1-p)[x>0][x\leq -1] + [x\leq 0] \\
&= p[x>1]y^{-x+1} + p[x=1] + (1-p)[x>0]y^{-x-1} + [x\leq 0] \\
&= p[x>1]y^{-x+1} + p[x=1]y^{-x+1} + (1-p)[x>0]y^{-x-1} + [x\leq 0] \\
&= p[x>0]y^{-x+1} + (1-p)[x>0]y^{-x-1} + [x\leq 0] \\
&= [x>0]\left(py^{-x+1} + (1-p)y^{-x-1}\right) + [x\leq 0] \\
&= [x>0]y^{-x}\left(py + (1-p)y^{-1}\right) + [x\leq 0]
\end{aligned}$$

In order for the last term to be at most I_p it suffices that $py + (1-p)y^{-1} = 1$. Solving this quadratic equation for y gives

$$y = 1 \;\vee\; y = \frac{1}{p} - 1\ .$$

The first solution gives the trivial superinvariant $I_1 = 1$ for all $p \in [0,1]$. The second solution gives the more interesting superinvariant

$$I_p = [x>0]\left(\frac{1}{p} - 1\right)^{-x} + [x\leq 0]$$

but only for $p \in (0,1]$ (for $p = 0$, we have of course the superinvariant $I_0 = [x \leq 0]$ because in this case, the program obviously terminates with probability 1 if $x \leq 0$, and with probability 0 if $x > 0$). From I_p we can infer, for instance, that for $p = 1/100$ the loop terminates only with probability at most 99^{-x} from initial state x.

Remark: It can be shown that P terminates with probability 1 iff $p \geq 1/2$. Note that our superinvariants do not contradict this fact.

(c)

$$[100>0]\left(\frac{1}{0.49} - 1\right)^{-100} + [100\leq 0] \;=\; \left(\frac{1}{0.49} - 1\right)^{-100} \;=\; \left(\frac{49}{51}\right)^{100} \;\leq\; 0.01831\ .$$

Exercise 2 (Exact *wp* via conditional difference boundedness) **30P**

Consider the following pGCL program P:

```
while (x > 0) {
    c := c + 1 ;
    { x := x − 1 } [1/2] { skip }
}
```

Determine $wp[\![P]\!](c)$ exactly. Proceed as follows:

- Guess a fixed point I of Φ_c and verify it.
- Verify that $I \sqsubseteq \text{lfp } \Phi_c$ by applying the rule from Slide 24 of Lecture 11 (this will only work if your guessed I is actually the least fixed point). You may assume without proof that P terminates in finite expected time from any initial state.

Hint: $wp[\![P]\!](c)$ is an expectation that describes the expected value of variable c after program termination given the initial values of x and c. Use this intuition to guess the correct I.

Solution: Intuitively, the loop takes $2x$ steps in expectation to terminate. So the expected value of c increases by $2x$ in expectation.
Let $I = c + 2x$.
We verify that I is a fixed point of Φ_f (remember to read this from bottom to top):

```
/// I                                   (definition)
/// c + 2x                              (simplify)
/// [x <= 0]c + [x > 0](c + 2x)         (apply Φ for the loop)

/// c + 2x
/// c + 1 + 2x − 1
c := c + 1
/// c + 2x − 1
{
  /// c + 2x
  skip
  /// c + 2x
} [1/2] {
  /// c + 2x − 2
  x := x - 1
  /// c + 2x
}
/// c + 2x(= I)
```

It only remains to show that I is conditionally difference bounded wrt. the loop body. So we have to check that $wp[\![P']\!](|I(s) - I|)(s) \leq c$ holds for all states s and some fixed constant $c \in \mathbb{R}_{\geq 0}$. Let s be an arbitrary but fixed program state:

/// 1
/// $0.5 \cdot |-1| + 0.5 \cdot |1|$
/// $0.5 \cdot |c(s) + 2x(s) - (c+1)(s) - 2x(s)| + 0.5 \cdot |c(s) + 2x(s) - (c+1)(s) - 2x(s) + 2|$
(we're done with the wp calculation and the expectation is evaluated in the input state, i.e. s)

/// $0.5 \cdot |(c+2x)(s) - (c+1) - 2x| + 0.5 \cdot |(c+2x)(s) - (c+1) - 2x + 2|$
`c := c + 1`
/// $0.5 \cdot |(c+2x)(s) - c - 2x| + 0.5 \cdot |(c+2x)(s) - c - 2x + 2|$
`{`
  /// $|(c+2x)(s) - c - 2x|$
  `skip`
  /// $|(c+2x)(s) - c - 2x|$
`} [1/2] {`
  /// $|(c+2x)(s) - c - 2x + 2|$
  /// $|(c+2x)(s) - c - 2(x-1)|$
  `x := x - 1`
  /// $|(c+2x)(s) - c - 2x|$
`}`
/// $|(c+2x)(s) - c - 2x|$
/// $|(c+2x)(s) - (c+2x)|$
/// $|I(s) - I|$

We see that for all states s we have $wp[\![P']\!](|I(s) - I|)(s) \leq 1$.

Exercise 3 (Conditioning in `pGCL` programs) **30P**

Consider the following scenario: A telephone operator has forgotten what day of the week it is. However, she knows that she receives on average ten calls per hour in the week and three calls per hour at the weekend. She observes that she receives four calls in a given hour.

(a) [10P] Write a `cpGCL` program P modeling the above scenario.
Hint: You may use a sampling statement like $x :\approx \mathcal{D}(p)$, where $\mathcal{D}$ is some discrete distribution and p a parameter.

Solution: Variable x encodes whether it is a weekday ($x = 0$) or it is weekend ($x = 1$).

Then program P is defined as follows:

$$
\begin{array}{ll}
1: & \{x := 0\}\,[5/7]\,\{x := 1\}; \\
2: & \texttt{if}(x = 0)\{r := 10\}\,\texttt{else}\,\{r := 3\}; \\
3: & d :\approx \text{poisson}(r); \\
4: & \texttt{observe}(d = 4);
\end{array}
$$

(b) [5P] Give an expectation f stating (for the program P) that it is a week day.

Solution: $f = [x = 0]$.

(c) [15P] Use the *cwp*-calculus to help the telephone operator to decide whether it is a week day. To this end, determine the probability that is a week day by computing $\mathit{cwp}(P, f)$.

Solution: We have to compute $cwp(P, [x=0]) = \frac{wp(P,[x=0])}{wlp(P,1)}$.

$$
\begin{aligned}
& wp(P, [x=0]) \\
= \;& wp(P_1; \texttt{observe}(d=4), [x=0]) \\
= \;& wp(P_1, wp(\texttt{observe}(d=4), [x=0])) \\
= \;& wp(P_1, [d=4] \cdot [x=0]) \\
= \;& wp(P_2; d := \sum_{k=0}^{\infty} \frac{r^k}{k! \cdot e^r} \cdot [k\rangle, [d=4] \cdot [x=0]) \\
= \;& wp(P_2, \texttt{wp}(d := \sum_{k=0}^{\infty} \frac{r^k}{k! \cdot e^r} \cdot [k\rangle, [d=4] \cdot [x=0])) \\
= \;& wp(P_2, \sum_{k=0}^{\infty} \frac{r^k}{k! \cdot e^r} \cdot [k=4][x=0]) \\
= \;& wp(P_2, \frac{r^4}{4! \cdot e^r} \cdot [x=0]) \\
= \;& wp(P_3; \texttt{if}(x=0)\{r := 10\}\texttt{else}\{r := 3\}, \frac{r^4}{4! \cdot e^r} \cdot [x=0]) \\
= \;& wp(P_3, wp(\texttt{if}(x=0)\{r := 10\}\texttt{else}\{r := 3\}, \frac{r^4}{4! \cdot e^r} \cdot [x=0])) \\
= \;& wp(P_3, [x=0] \cdot wp(r := 10, \frac{r^4}{4! \cdot e^r} \cdot [x=0]) + [x \neq 0] \cdot wp(r := 3, \frac{r^4}{4! \cdot e^r} \cdot [x=0])) \\
= \;& wp(P_3, [x=0] \cdot \frac{10^4}{4! \cdot e^{10}} \cdot [x=0] + [x \neq 0] \cdot \frac{3^4}{4! \cdot e^3} \cdot [x=0]) \\
= \;& wp(P_3, [x=0] \cdot \frac{10^4}{4! \cdot e^{10}}) \\
= \;& wp(\{x := 0\}[5/7]\{x := 1\}, [x=0] \cdot \frac{10^4}{4! \cdot e^{10}}) \\
= \;& 5/7 \cdot wp(x := 0, [x=0] \cdot \frac{10^4}{4! \cdot e^{10}}) + 2/7 \cdot wp(x := 1, [x=0] \cdot \frac{10^4}{4! \cdot e^{10}}) \\
= \;& 5/7 \cdot [0=0] \cdot \frac{10^4}{4! \cdot e^{10}} + 2/7 \cdot [1=0] \cdot \frac{10^4}{4! \cdot e^{10}} \\
= \;& 5/7 \cdot \frac{10^4}{4! \cdot e^{10}} \;\approx\; 0.0135.
\end{aligned}
$$

Solution: Next, we compute the weakest liberal pre-expectation:

$$
\begin{aligned}
& wlp(P, 1) \\
= \;& wlp(P_1; \texttt{observe}(d = 4), 1) \\
= \;& wlp(P_1, wlp(\texttt{observe}(d = 4), 1)) \\
= \;& wlp(P_1, [d = 4]) \\
= \;& wlp(P_2; d := \sum_{k=0}^{\infty} \frac{r^k}{k! \cdot e^r} \cdot [k\rangle, [d = 4]) \\
= \;& wlp(P_2, \texttt{wlp}(d := \sum_{k=0}^{\infty} \frac{r^k}{k! \cdot e^r} \cdot [k\rangle, [d = 4])) \\
= \;& wlp(P_2, \sum_{k=0}^{\infty} \frac{r^k}{k! \cdot e^r} \cdot [k = 4]) \\
= \;& wlp(P_2, \frac{r^4}{4! \cdot e^r}) \\
= \;& wlp(P_3; \texttt{if}(x = 0)\{r := 10\}\texttt{else}\{r := 3\}, \frac{r^4}{4! \cdot e^r}) \\
= \;& wlp(P_3, wlp(\texttt{if}(x = 0)\{r := 10\}\texttt{else}\{r := 3\}, \frac{r^4}{4! \cdot e^r})) \\
= \;& wlp(\{x := 0\}[5/7]\{x := 1\}, [x = 0] \cdot \frac{10^4}{4! \cdot e^{10}} + [x \neq 0] \cdot \frac{3^4}{4! \cdot e^3}) \\
= \;& 5/7 \cdot \frac{10^4}{4! \cdot e^{10}} + 2/7 \cdot \frac{3^4}{4! \cdot e^3} \\
\approx \;& 0.0615
\end{aligned}
$$

It remains to combine both results to obtain the conditional weakest pre-expectation:

$$
\begin{aligned}
& cwp(P, [x = 0]) \\
= \;& \frac{wp(P, [x = 0])}{wlp(P, 1)} \\
= \;& \frac{5/7 \cdot \frac{10^4}{4! \cdot e^{10}}}{5/7 \cdot \frac{10^4}{4! \cdot e^{10}} + 2/7 \cdot \frac{3^4}{4! \cdot e^3}} \\
\approx \;& 0.2196 \; .
\end{aligned}
$$

Hence, the probability that it is a week day given the operator's observations is roughly 0.2196.

Hint: We assume that probability of receiving k calls in an hour is given by a (discrete) poisson distribution. That is, if we know that we receive on average r calls per hour, then the probability of receiving k calls in a given hour is $\frac{r^k}{k!} \cdot e^{-r}$.

Exercise 4 (From pGCL to conditional reward Markov chains) **30P**

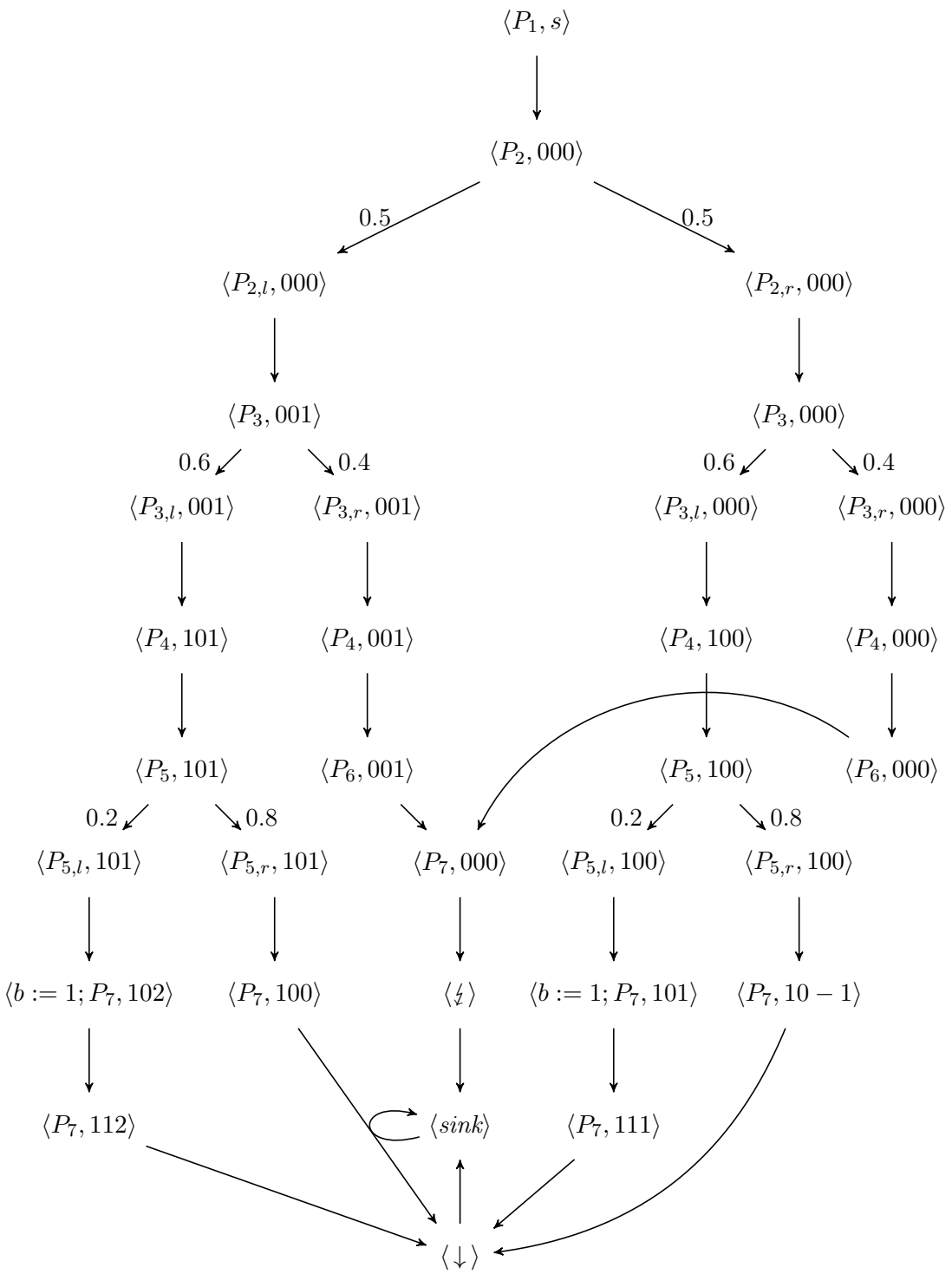
Consider the following pGCL program P:

$$
\begin{array}{ll}
1: & a, b, c := 0, 0, 0; \\
2: & \{c := c + 1;\}\ [0.5]\ \{\mathsf{skip}\}; \\
3: & \{a := 1\}\ [0.6]\ \{\mathsf{skip}\}; \\
4: & \mathsf{if}(a = 1)\{ \\
5: & \qquad \{c := c + 1; b := 1\}\ [0.2]\ \{c := c - 1\}; \\
6: & \}\ \mathsf{else}\ \{ \\
7: & \qquad c := 0 \\
8: & \}; \\
9: & \mathsf{observe}\ (a \neq 0\ \vee\ b \neq 0)
\end{array}
$$

(a) [20P] Construct the reward Markov chain corresponding to P (you may use the line numbers to refer to sub-programs).

(b) [10P] Compute the expected value of c after termination of P.

Solution: We write P_i to denote the sub-program of P that starts with line i. Further, $P_{i,l}$ and $P_{i,r}$ denote the left and right branch of such a program in case of probabilistic choice or conditional statements. We denote program states as follows: We write 123 instead of $s[a \mapsto 1, b \mapsto 2, c \mapsto 3]$.



We assign a reward corresponding to the value of c to every state with first component P_7. All other states have a reward of 0.

Then the expected value of c after eventually reaching $\langle sink \rangle$ under the condition $\neg\langle \lightning \rangle$ is

$$\begin{aligned} ExpRew(\Diamond\langle sink\rangle \mid \neg\langle\lightning\rangle) &= \frac{ExpRew(\Diamond\langle sink\rangle \cap \neg\Diamond\langle\lightning\rangle)}{Pr(\neg\Diamond\langle\lightning\rangle)} \\ &= \frac{0.5\cdot 0.6\cdot 0.2\cdot 2 + 0.5\cdot 0.6\cdot 0.2\cdot 1 + 0.5\cdot 0.6\cdot 0.8\cdot(-1)}{1-(0.5\cdot 0.4 + 0.5\cdot 0.4)} \\ &= \frac{0.12+0.06-0.24}{0.6} \;=\; -0.1 \end{aligned}$$

Exercise 5 (Guard strengthening for lower bounds) **30P**

Prove the "guard strengthening for lower bounds" proof rule from lecture 11, slide 26:

> Let $P_{loop} = \texttt{while}\ (\varphi)\ \{\ P\ \}$ and $P'_{loop} = \texttt{while}\ (\varphi')\ \{\ P\ \}$, and expectations f and I. Then it holds:
>
> $$\big(\varphi' \Rightarrow \varphi \quad\wedge\quad I \sqsubseteq wp[\![P'_{loop}]\!]([\neg\varphi]\cdot f)\big) \quad \text{implies} \quad I \sqsubseteq wp[\![P_{loop}]\!](f)\ .$$

Hint 1: Prove $wp[\![P'_{loop}]\!]([\neg\varphi]\cdot f) \sqsubseteq wp[\![P_{loop}]\!](f)$ where $\varphi' \Rightarrow \varphi$ first.

Hint 2: In exercise sheet 4, exercise 3 (a), you have shown the following useful lemma:

> Let $(D, \sqsubseteq)$ be a complete lattice and $f, g\colon D \to D$ be monotonic such that for all $d \in D$, $f(d) \sqsubseteq g(d)$. Then, lfp $f \sqsubseteq$ lfp g holds.

Solution: Let $P_{loop} = \texttt{while}\ (\varphi)\ \{\ P\ \}$ and $P'_{loop} = \texttt{while}\ (\varphi')\ \{\ P\ \}$ where $\varphi' \Rightarrow \varphi$, and expectations f and I where $I \sqsubseteq wp[\![P'_{loop}]\!]([\neg\varphi]\cdot f)$.

We show $wp[\![P'_{loop}]\!]([\neg\varphi]\cdot f) \sqsubseteq wp[\![P_{loop}]\!](f)$ first.
From the definition of wp, we obtain the following:

$$\begin{aligned} & wp[\![P'_{loop}]\!]([\neg\varphi]\cdot f) \sqsubseteq wp[\![P_{loop}]\!](f) \\ \text{iff} \quad & \text{lfp}\ \Phi'_{[\neg\varphi]\cdot f} \sqsubseteq \text{lfp}\ \Phi_f \end{aligned}$$

We can show that the latter equation holds using the second hint's lemma. From the lecture, we know that $\Phi'_{[\neg\varphi]\cdot f}$ and Φ_f are monotonic. We prove that $\Phi'_{[\neg\varphi]\cdot f} \sqsubseteq \Phi_f$ holds. For all $X \in \mathbb{E}$:

$$\begin{aligned} & \Phi'_{[\neg\varphi]\cdot f}(X) \sqsubseteq \Phi_f(X) \\ \text{iff} \quad & [\varphi']\cdot wp[\![P]\!](X) + [\neg\varphi']\cdot([\neg\varphi]\cdot f) \sqsubseteq [\varphi]\cdot wp[\![P]\!](X) + [\neg\varphi]\cdot f \qquad \text{(def. } \Phi\text{)} \end{aligned}$$

We show the above inequality for each state $s \in \mathbb{S}$:

$$([\varphi']\cdot wp[\![P]\!](X) + [\neg\varphi']\cdot([\neg\varphi]\cdot f))(s) \le ([\varphi]\cdot wp[\![P]\!](X) + [\neg\varphi]\cdot f)(s)$$

If $[\varphi'](s) = 1$ holds, then also $[\varphi](s) = 1$ holds by the lemma's first assumption. Then, we have:

$$\begin{aligned} & (1\cdot wp[\![P]\!](X) + 0\cdot(0\cdot f))(s) \le (1\cdot wp[\![P]\!](X) + 0\cdot f)(s) \\ \text{iff} \quad & wp[\![P]\!](X)(s) \le wp[\![P]\!](X)(s) \end{aligned}$$

On the other hand, if $[\varphi'](s) = 0$ holds:

$$\begin{aligned} & (0 \cdot wp[\![P]\!](X) + 1 \cdot ([\neg\varphi] \cdot f))(s) \leq ([\varphi] \cdot wp[\![P]\!](X) + [\neg\varphi] \cdot f)(s) \\ \text{iff} \quad & ([\neg\varphi] \cdot f)(s) \leq ([\varphi] \cdot wp[\![P]\!](X) + [\neg\varphi] \cdot f)(s) \end{aligned}$$

This concludes the proof for $wp[\![P'_{loop}]\!]([\neg\varphi] \cdot f) \sqsubseteq wp[\![P_{loop}]\!](f)$.
By $I \sqsubseteq wp[\![P'_{loop}]\!]([\neg\varphi] \cdot f)$, it follows that $I \sqsubseteq wp[\![P_{loop}]\!](f)$ holds.