Prof. Dr. Ir. Dr. h.c. Joost-Pieter Katoen
Lutz Klinkenberg, Philipp Schröer, Tobias Winkler

Exercise Sheet 7

General remarks:

- **Due date:** Thursday, December 15$^{\text{th}}$ 16:30 (before the exercise class).
- 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 (The Arithmetical Hierarchy) **20P**

Consider the following decision problem *INF*:

- Input: A (non-probabilistic) GCL program P with a single non-negative integer variable $\mathtt{v}$.
- Output: *Yes*, if P terminates for infinitely many initial values of $\mathtt{v}$; *No*, otherwise.

Identify a class A of the arithmetical hierarchy such that *INF* is A-complete. Prove that your answer is correct.

Solution: We claim that *INF* is Π_2-complete.

- To show that $\mathit{INF} \in \Pi_2$ we argue as follows: For a GCL program P, we write

$$\mathcal{W}_P \;:=\; \{s \in \mathbb{Z}_{\geq 0} \mid (P, s) \in H, \text{ i.e., } P \text{ terminates on initial state } \mathtt{v} = s\} \;.$$

Note that $\mathit{INF} = \{P \in \mathrm{GCL} \mid |\mathcal{W}_P| = \infty\}$.

$$\begin{aligned}
& P \in \mathit{INF} \\
\iff\; & |\mathcal{W}_P| = \infty \\
\iff\; & \forall s \exists s' \colon s' > s \;\wedge\; s' \in \mathcal{W}_P \\
\iff\; & \forall s \exists s' \colon s' > s' \;\wedge\; (P, s') \in H \\
\iff\; & \forall s \exists s' \exists k \colon s' > s \;\wedge\; P \text{ terminates on input } s' \text{ in } k \text{ steps} \;,
\end{aligned}$$

which is a Π_2-formula.

- To show that *INF* is Π_2-hard, we reduce from the universal halting problem *UH*. Since $\mathit{UH} \in \Pi_2$ there exists a decidable relation $R(x, y, z)$ such that

$$P \in \mathit{UH} \quad\iff\quad \forall y \exists z \colon R(P, y, z) \;.$$

Now suppose that we are given an instance of UH, i.e., a GCL program P. From P, we construct an instance of INF – another program P' – as follows: Given an initial state $\mathtt{v} = s$, P' simulates P on all (finitely many) initial states $\mathtt{v} = y$ for $y \leq s$ *in parallel*. Then

$$(P', s) \in H \iff \forall y \leq s\ \exists z\colon R(P, y, z)\ ,$$

i.e., P' terminates with input $\mathtt{v} = s$ if and only if P terminates on all inputs $\mathtt{v} = y$ for $y \leq s$. Therefore:

$$\begin{aligned}
& P' \in INF \\
\iff\ & |\mathcal{W}_{P'}| = \infty \\
\iff\ & \forall s \exists s'\colon s' > s\ \wedge\ s' \in \mathcal{W}_{P'} \\
\iff\ & \forall s \exists s'\colon s' > s\ \wedge\ (P', s') \in H \\
\iff\ & \forall s \exists s'\colon s' > s\ \wedge\ \forall y \leq s'\ \exists z\colon R(P, y, z) \\
\iff\ & \forall y \exists z\colon R(P, y, z) \\
\iff\ & P \in UH
\end{aligned}$$

Exercise 2 (Proving Almost-Sure Termination) **35P**

Consider the pGCL program P below:

```
while (x ≠ 10) {
    if (x is even) {
        {x := x − 2} [1/2] {x := x + 2}
    } else {
        x := x + 1
    }
}
```

Here, we assume that x is an integer variable. Use the proof rule for almost-sure termination from Lecture #15 (the rule involving the antitone functions p and d) to show that P terminates almost-surely for any given initial value of x.

Hint: Consider the expectation $V = 3 \cdot [x \text{ is odd}] + |x - 10|$ and choose *constant* functions p and d.

Solution: We choose $I = \mathtt{true}$, $V = 3 \cdot [x \text{ is odd}] + |x - 10|$, $p = 1/2$, and $d = 2$.
Clearly, both p and d are antitone as they are constant functions. It then remains to check the four premises of the proof rule:

1. $[I]$ is a *wp*-subinvariant of P with respect to $[I]$: Let Φ be the characteristic function of

the loop of program P. $[I] = [\texttt{true}] = 1$. Let P' denote the loop body of P. We have:

$$
\begin{aligned}
\Phi_1(1) &= [x = 10] \cdot 1 + [x \neq 10] \cdot wp(P', 1) \\
&= [x = 10] + [x \neq 10] \cdot ([x \text{ is even}] \cdot \underbrace{wp(\{x := x-2\}[1/2]\{x := x+2\}, 1)}_{=1} \\
&\quad + [x \text{ is odd}] \cdot \underbrace{wp(x := x+1, 1)}_{=1}) \\
&= [x = 10] + [x \neq 10] \cdot ([x \text{ is even}] \cdot 1 + [x \text{ is odd}] \cdot 1) \\
&= [x = 10] + [x \neq 10] \;=\; 1.
\end{aligned}
$$

2. $[\neg G] = [V = 0]$.
It is trivial that the negation of the guard (loop termination) leads to $V = 0$:

$$
\begin{aligned}
[V = 0] &= [(3 \cdot [x \text{ is odd}] + |x - 10|) = 0] \\
&= [[x \text{ is odd}] = 0 \text{ and } |x - 10| = 0] \\
&= [[x \text{ is even}] = 1 \text{ and } [x = 10] = 1] \\
&= [[x \text{ is even}] \cdot [x = 10]] \\
&= [x = 10] \\
&= [\neg(x \neq 10)] \;=\; [\neg G].
\end{aligned}
$$

3. V is a super-invariant of P with respect to V.
First, let compute $wp(P', V)$.

$$
\begin{aligned}
wp(P', V) &= [\text{x is even}] \cdot 1/2 \cdot (V[x := x-2] + V[x := x+2]) + [x \text{ is odd}] \cdot V[x := x+1] \\
&= [x \text{ is even}] \cdot 1/2 \cdot ((3 \cdot [(x-2) \text{ is odd}] + |x - 2 - 10|) \\
&\quad + (3 \cdot [(x+2) \text{ is odd}] + |x + 2 - 10|)) \\
&\quad + [x \text{ is odd}] \cdot (3 \cdot [(x+1) \text{ is odd}] + |x + 1 - 10|) \\
&\qquad\qquad (\text{We have } [x \text{ is even}] \cdot [x \pm 2 \text{ is odd}] = 0) \\
&\qquad\qquad (\text{We also have } [x \text{ is odd}] \cdot [x + 1 \text{ is odd}] = 0) \\
&= [x \text{ is even}] \cdot 1/2 \cdot (|x - 8| + |x - 12|) + \cdot [x \text{ is odd}] \cdot |x - 9|
\end{aligned}
$$

$$
\begin{aligned}
\Phi_V(V) &= [x = 10] \cdot V + [x \neq 10] \cdot wp(P', V) \\
&= [x = 10] \cdot (3 \cdot [x \text{ is odd}] + |x - 10|) \\
&\quad + [x \neq 10] \cdot ([x \text{ is even}] \cdot 1/2 \cdot (|x - 8| + |x - 12|) + [x \text{ is odd}] \cdot |x - 9|) \\
&\qquad\qquad (\text{We have } [x = 10] \cdot [x \text{ is odd}] = 0) \\
&= [x = 10] \cdot |x - 10| + [x \neq 10] \cdot ([x \text{ is even}] \cdot 1/2 \cdot (|x - 8| + |x - 12|) + [x \text{ is odd}] \cdot |x - 9|)
\end{aligned}
$$

We have $1/2 \cdot (|x - 8| + |x - 12|) \leq |x - 10|$, for $x \leq 8$ and $x \geq 12$ which covers all even numbers except 10, i.e., all the numbers fitting in $[x \neq 10] \cdot [x \text{ is even}]$. Therefore:

$$
\begin{aligned}
\Phi_V(V) \;&\leq\; [x = 10] \cdot |x - 10| + [x \neq 10] \cdot ([x \text{ is even}] \cdot |x - 10| + [x \text{ is odd}] \cdot |x - 9|) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(We have } |x-9| \leq |x-10|+1) \\
&\leq\; [x = 10] \cdot |x - 10| + [x \neq 10] \cdot ([x \text{ is even}] \cdot |x - 10| + [x \text{ is odd}] \cdot (|x - 10| + 1)) \\
&\leq\; [x = 10] \cdot |x - 10| + [x \neq 10] \cdot |x - 10| + [x \neq 10] \cdot [x \text{ is odd}] \\
&\leq\; |x - 10| + [x \neq 10] \cdot [x \text{ is odd}] \\
&\leq\; V.
\end{aligned}
$$

You can also argue by considering different cases:
For $x = 10$, $\Phi_V(V) = 0$ and $V = 0$, so $\Phi_V(V) \leq V$.
For $x = 9$ and $x = 11$, $\Phi_V(V) = 4$ and $V = 4$, so $\Phi_V(V) \leq V$.
For $x \leq 8$ and $x \geq 12$ and x even, $\Phi_V(V) \leq |x - 10|$ and $V = |x - 10|$, so $\Phi_V(V) \leq V$.
For $x \leq 8$ and $x \geq 12$ and x odd, $\Phi_V(V) \leq |x - 9|$ and $V = |x - 10| + 3$, so $\Phi_V(V) \leq V$.

4. V satisfies the progress condition

$$(p \circ V) \cdot [G] \cdot [I] \leq \lambda s.wp(P', [V \leq V(s) - d(V(s))])(s),$$

where P' is the loop body of P. Let $f_s = [V \leq V(s) - 2]$. Then:

$$
\begin{aligned}
&\lambda s.wp(P', f_s)(s) \\
=\;& \lambda s.\big(1/2 \cdot [x \text{ is even}] \cdot \big((3 \cdot [x - 2 \text{ is odd}] + |x - 2 - 10|) \leq (3 \cdot [x(s) \text{ is odd}] + |x(s) - 10|) - 2\big) \\
&\; + (3 \cdot [x + 2 \text{ is odd}] + |x + 2 - 10|) \leq (3 \cdot [x(s) \text{ is odd}] + |x(s) - 10|) - 2\big)\big)(s) \\
&\; + [x \text{ is odd}](3 \cdot [x + 1 \text{ is odd}] + |x + 1 - 10| \leq 3 \cdot [x(s) \text{ is odd}] + |x(s) - 10| - 2)\big) \\
\\
=\;& 1/2 \cdot [x \text{ is even}] \cdot \Big(\underbrace{(3 \cdot [x - 2 \text{ is odd}] + |x - 12|) \leq (3 \cdot [x \text{ is odd}] + |x - 10|) - 2}_{=\,1,\, for\, x \geq 12}\Big) \\
&\; + \underbrace{(3 \cdot [x + 2 \text{ is odd}] + |x - 8|) \leq (3 \cdot [x \text{ is odd}] + |x - 10|) - 2}_{=\,1,\, for\, x \leq 8} \\
&\; + [x \text{ is odd}]\, \underbrace{(3 \cdot \underbrace{[x + 1 \text{ is odd}]}_{=\,0} + |x - 9| \leq \underbrace{3 \cdot [x \text{ is odd}]}_{=\,3} + |x - 10| - 2)}_{=\,1\ (|x-9| \leq |x-10|+1)}\Big) \\
=\;& 1/2 \cdot [x \text{ is even}] + [x \text{ is odd}] \\
=\;& 1/2 \cdot [x \text{ is even}] + (1/2 + 1/2) \cdot [x \text{ is odd}] \\
=\;& 1/2 + 1/2 \cdot [x \text{ is odd}]
\end{aligned}
$$

We have:

$$
\begin{aligned}
&(p \circ V) \cdot [G] \cdot [I] \\
=\;& (1/2 \circ V) \cdot [x \neq 10] \cdot [\mathtt{true}] \\
=\;& 1/2 \cdot [x \neq 10] \\
\leq\;& 1/2 + 1/2 \cdot [x \text{ is odd}] = \lambda s.wp(P', f_s)(s).
\end{aligned}
$$

Since all four premises of the proof rule are satisfied, we conclude that

$$wp(P, 1) \;\geq\; [true] \;=\; 1.$$

In other words, P terminates almost-surely.

Exercise 3 (Positive Almost-Sure Termination) $\boxed{\textbf{20P}}$

Consider a pGCL program P of the form

$$\texttt{while}\,(G)\,\{P'\}\ ,$$

where P' is a loop-free pGCL program. A clever student suggests the following scheme to prove positive almost-sure termination by weakest preexpectation reasoning:

1. Modify program P by introducing a fresh variable, say v, which is initialized with 0.
2. Increment v for every loop iteration by 1.

Hence, the modified program $\hat{P}$ is given by

$$v := 0;\ \texttt{while}\,(G)\,\{v := v + 1; P'\}.$$

Prove or disprove: $wp(\hat{P}, v)(s) < \infty$ implies that P terminates positive almost-surely on initial state s.

Solution: Consider the following program P:

$$\texttt{while(true)}\ \{\ \texttt{skip}\ \}$$

Since this program never terminates, it also does not terminate almost-surely. In fact, we have already shown in exercise 1 (c) on exercise sheet 5 that $wp(P, 1) = 0$ holds. Now, consider the corresponding transformed program $\hat{P}$:

$$v := 0; \texttt{while(true)}\ \{\ v := v + 1; \texttt{skip}\ \}$$

We claim that $I = 0$ is a suitable invariant for the loop of this program w.r.t. v:

$$\begin{aligned}\Phi_v(I) \;&=\; [\texttt{false}] \cdot v + [\texttt{true}] \cdot wp(v := v + 1; \texttt{skip})(I) \\ &=\; 0 + 1 \cdot I[v := v + 1] \;=\; 0 \leq I.\end{aligned}$$

Hence, $wp(\hat{P}, v) = 0 < \infty$, but P does not terminate almost surely *for any state s*.

Aside: We have a conjecture that such a transformation does indeed allow proving positive almost-sure termination on programs that are already almost-surely terminating. We believe that it is true, but over the last several years no one has presented a formal proof.