Exercise session 10
– Extensions of IMP –

Exercise 34. Locations

Consider the following statement:

```
begin
  var x:=0;
  proc p begin x:=x+1 end;
  proc q begin call p end;
  begin
    proc p begin x:=7 end;
    call q
  end
end
```

Using the extended semantics of **IMP**, show that the final store will associate the location of `x` with value 1.

Exercise 35. Continuation

Determine the semantics of the following statement:

```
begin
  while true do
    if x <= 0 then raise exit
              else x := x-1  end
  end
  handle exit: y:=7
end
```

Solutions

Exercise 34. Locations

We know that before executing the statement $\Phi_V = \{\}$, $\Phi_P = \{\}$, and $\$ = \{\texttt{next} \mapsto 0\}$.
In order to calculate the final store, we have to use the extended semantics on the statement:

$$\mathcal{S}'_{DS}[\![\texttt{begin } \underbrace{\texttt{var x:=0;}}_{D_V}\ \underbrace{\texttt{proc p begin x:=x+1 end; proc q begin call p end;}}_{D_P}\texttt{ s end}]\!](\Phi_V, \Phi_P, \$)$$

where s is the inner block: `begin proc p begin x:=7 end; call q end`.
This gives:

$$\begin{aligned}
& \mathcal{S}'_{DS}[\![\texttt{begin var x:=0; proc p begin x:=1 end; proc q begin call p end; s end}]\!](\Phi_V, \Phi_P, \$) \\
= \ & \mathcal{S}'_{DS}[\![\texttt{s}]\!](\Phi'_V, \Phi'_P, \$') \\
& \text{where } (\Phi'_V, \$') = D^V_{DS}[\![D_V]\!](\Phi_V, \$) \\
& \qquad = (\Phi_V[\texttt{x} \mapsto \$(\texttt{next})], \$[\$(\texttt{next}) \mapsto \mathcal{A}[\![0]\!](\mathit{lookup}(\Phi_V, \$))][\texttt{next} \mapsto \mathit{new}(\$(\texttt{next}))]) \\
& \qquad = (\Phi_V[\texttt{x} \mapsto 0], \$[0 \mapsto 0][\texttt{next} \mapsto 1]) \\
& \text{and } \Phi'_P = D^P_{DS}[\![D_P]\!](\Phi'_V, \Phi_P) \\
& \qquad = D^P_{DS}[\![\texttt{proc q begin call p end}]\!](\Phi'_V, \Phi_P[\texttt{p} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=x+1}]\!](\Phi'_V, \Phi_P)]) \\
& \qquad = \Phi_P[\texttt{p} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=x+1}]\!](\Phi'_V, \Phi_P)][\texttt{q} \mapsto \mathcal{S}'_{DS}[\![\texttt{call p}]\!](\Phi'_V, \Phi_P[\texttt{p} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=x+1}]\!](\Phi'_V, \Phi_P)])] \\
& \qquad = \Phi_P[\texttt{p} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=x+1}]\!](\Phi'_V, \Phi_P)][\texttt{q} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=x+1}]\!](\Phi'_V, \Phi_P)] \\
= \ & \mathcal{S}'_{DS}[\![\texttt{begin proc p begin x:=7 end; call q end}]\!](\Phi'_V, \Phi'_P, \$') \\
= \ & \mathcal{S}'_{DS}[\![\texttt{call q}]\!](\Phi'_V, \Phi'_P[\texttt{p} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=7}]\!](\Phi'_V, \Phi'_P)], \$') \\
= \ & (\Phi'_P[\texttt{p} \mapsto \mathcal{S}'_{DS}[\![\texttt{x:=7}]\!](\Phi'_V, \Phi'_P)](\texttt{q}))(\$') \\
= \ & \mathcal{S}'_{DS}[\![\texttt{x:=x+1}]\!](\Phi'_V, \Phi_P, \$') \\
= \ & \$'[\Phi'_V(\texttt{x}) \mapsto \mathcal{A}[\![\texttt{x}]\!](\mathit{lookup}(\Phi'_V, \$')) + 1] = \$'[0 \mapsto 0 + 1] = \$'[0 \mapsto 1]
\end{aligned}$$

To get the final value of `x`, we use the lookup function in the final variable environment and store:
$\Phi'_V = \Phi_V[\texttt{x} \mapsto 0] = \{\texttt{x} \mapsto 0\}$,
$\$'[0 \mapsto 1] = \{0 \mapsto 1, \texttt{next} \mapsto 1\}$.

$$\mathit{lookup}(\{\texttt{x} \mapsto 0\}, \{0 \mapsto 1, \texttt{next} \mapsto 1\})(\texttt{x}) = 1.$$

Exercise 35. Continuation

We have to determine

$$\mathcal{S}_{CS}[\![\texttt{begin } s_1 \texttt{ handle exit : } s_2 \texttt{ end}]\!](\Phi_E, c)$$

where s_1 is the `while` statement,
s_2 is statement `y:=7`,
Φ_E the empty exception environment $\{\}$,
c the id function on $State$.

Using the extended semantics, we get:

$$\begin{aligned}
& \mathcal{S}_{CS}[\![\texttt{begin } s_1 \texttt{ handle exit : } s_2 \texttt{ end}]\!](\{\}, id) \\
= \ & \mathcal{S}_{CS}[\![s_1]\!](\underbrace{\{\texttt{exit} \mapsto \mathcal{S}_{CS}[\![\texttt{y:=7}]\!](\{\}, id)\}}_{\Phi'_E}, id) \\
= \ & FIX\ F
\end{aligned}$$

where

$$F(g)c \ = \ cond(true, \mathcal{S}_{CS}[\![s_3]\!](\Phi'_E, g(c)), id) \ = \ \mathcal{S}_{CS}[\![s_3]\!](\Phi'_E, g(c))$$

where s_3 is the `if` statement inside the `while` statement.

Now we have to determine $F(g)c$ (with $c = id$) and do the fixed point iteration on it:

$$\begin{aligned}
(F(g)id)\sigma \ &= \mathcal{S}_{CS}[\![s_3]\!](\Phi'_E, g(id), \sigma) \\
&= cond(\mathcal{B}[\![\texttt{x<=0}]\!], \mathcal{S}_{CS}[\![\texttt{raise exit}]\!](\Phi'_E, g(id)), \mathcal{S}_{CS}[\![\texttt{x:=x} - 1]\!](\Phi'_E, g(id)))\sigma \\
&= \begin{cases} \mathcal{S}_{CS}[\![\texttt{raise exit}]\!](\Phi'_E, g(id))\sigma & \text{if } \sigma(\texttt{x}) \leq 0 \\ \mathcal{S}_{CS}[\![\texttt{x:=x} - 1]\!](\Phi'_E, g(id))\sigma & \text{if } \sigma(\texttt{x}) > 0 \end{cases} \\
&= \begin{cases} \mathcal{S}_{CS}[\![\texttt{y:=7}]\!](\{\}, id)\sigma & \text{if } \sigma(\texttt{x}) \leq 0 \\ g(id)(\sigma[\texttt{x} \mapsto \sigma(\texttt{x}) - 1]) & \text{if } \sigma(\texttt{x}) > 0 \end{cases} \\
&= \begin{cases} id(\sigma[\texttt{y} \mapsto 7]) & \text{if } \sigma(\texttt{x}) \leq 0 \\ g(id)(\sigma[\texttt{x} \mapsto \sigma(\texttt{x}) - 1]) & \text{if } \sigma(\texttt{x}) > 0 \end{cases} \\
&= \begin{cases} \sigma[\texttt{y} \mapsto 7] & \text{if } \sigma(\texttt{x}) \leq 0 \\ g(id)(\sigma[\texttt{x} \mapsto \sigma(\texttt{x}) - 1]) & \text{if } \sigma(\texttt{x}) > 0 \end{cases}
\end{aligned}$$

Now we can start the fixed point iteration:

$$(F(\perp)id)\sigma \ = \begin{cases} \sigma[\texttt{y} \mapsto 7] & \text{if } \sigma(\texttt{x}) \leq 0 \\ \text{undefined} & \text{if } \sigma(\texttt{x}) > 0 \end{cases}$$

$$(F^2(\bot)id)\sigma \quad = \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ (F(\bot)id)\sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 1] & \text{if } \sigma(\mathtt{x}) > 0 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 1][\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) > 0 \wedge \sigma(\mathtt{x}) \leq 1 \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) > 0 \wedge \sigma(\mathtt{x}) > 1 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 1][\mathtt{y} \mapsto 7] & \text{if } 0 < \sigma(\mathtt{x}) \leq 1 \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) > 1 \end{cases}$$

$$(F^3(\bot)id)\sigma \quad = \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ (F^2(\bot)id)\sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 1] & \text{if } \sigma(\mathtt{x}) > 0 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 1][\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) > 0 \wedge \sigma(\mathtt{x}) \leq 1 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 2][\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) > 0 \wedge 1 < \sigma(\mathtt{x}) \leq 2 \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) > 0 \wedge \sigma(\mathtt{x}) > 2 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 1][\mathtt{y} \mapsto 7] & \text{if } 0 < \sigma(\mathtt{x}) \leq 1 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - 2][\mathtt{y} \mapsto 7] & \text{if } 1 < \sigma(\mathtt{x}) \leq 2 \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) > 2 \end{cases}$$

Now we can see the pattern of the iteration:

$$(F^n(\bot)id)\sigma = \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - j][\mathtt{y} \mapsto 7] & \text{if } j - 1 < \sigma(\mathtt{x}) \leq j \;\wedge\; 0 < j < n \\ \text{undefined} & \text{if } \sigma(\mathtt{x}) > n - 1 \end{cases}$$

We can observe that the third case can be dropped when determining the fixed point, because n can be arbitrary big, thus there is no value $\sigma(\mathtt{x})$ that can be greater than the right-hand side.
Thus, the fixed point and our final result is:

$$((FIX\ F)id)\sigma \quad = \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - n][\mathtt{y} \mapsto 7] & \text{if } n - 1 < \sigma(\mathtt{x}) \leq n \;\wedge\; n > 0 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - n][\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) = n \;\wedge\; n > 0 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto \sigma(\mathtt{x}) - \sigma(\mathtt{x})][\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) > 0 \end{cases}$$

$$= \begin{cases} \sigma[\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) \leq 0 \\ \sigma[\mathtt{x} \mapsto 0][\mathtt{y} \mapsto 7] & \text{if } \sigma(\mathtt{x}) > 0 \end{cases}$$