# Assignment 4: Solution

## Exercise 1

See the file `counter.als`.

## Exercise 2

See the files `imagefile_eager.als` and `imagefile_lazy.als`.

## Exercise 3

The model defines a set of objects *Node* and one relation $next \subseteq Node \times Node$.

The given model has one constraint $c$: for every node $n$ there is exactly one node $m$ such that $(n, m) \in next$. The assertion $a$ checks whether for every node $n$ there exists a node $m$ with $(m, n) \in next$.

Given two nodes $n$ and $m$, we will write $x_{n,m}$ to denote $(n, m) \in next$.

1. For the scope with one object, we have $Node = \{0\}$.

   We encode the constraint $c$ as $x_{0,0}$.

   We encode the assertion $a$ as $x_{0,0}$.

   The resulting boolean formula is $x_{0,0} \wedge \neg x_{0,0}$. This formula is not satisfiable. Therefore, for the given scope there is no counter-example for the assertion.

2. For the scope with two objects, we have $Node = \{0, 1\}$.

   We encode the constraint as

   $$c := ((x_{0,0} \wedge \neg x_{0,1}) \vee (\neg x_{0,0} \wedge x_{0,1})) \wedge ((x_{1,0} \wedge \neg x_{1,1}) \vee (\neg x_{1,0} \wedge x_{1,1})) \ .$$

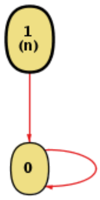
   We encode the assertion as $a := (x_{0,0} \vee x_{1,0}) \wedge (x_{0,1} \vee x_{1,1})$.

   The resulting boolean formula is

   $$c \wedge \neg a \ .$$

The boolean formula is satisfied when

$$x_{0,0} = T \quad x_{0,1} = F \quad x_{1,0} = T \quad x_{1,1} = F\ .$$

The counter-example can be visualized as 

.

3. The new field and fact result in two additional constraints: ($c_1$) every node has exactly one previous node, and ($c_2$) for every node $n$, there exists a node $m$ such that $(n, m) \in \mathit{next}, (m, n) \in \mathit{prev}$. We will write $p_{n,m}$ to denote $(n, m) \in \mathit{prev}$.

   **Scope 1:** For checking `check demo for 1` we encode the constraints as:

$$\begin{aligned} x_{0,0} & \qquad (\mathit{Constraint}\ c) \\ p_{0,0} & \qquad (\mathit{Constraint}\ c_1) \\ x_{0,0} \wedge p_{0,0} & \qquad (\mathit{Constraint}\ c_2) \end{aligned}$$

   and the assertion is encoded as before:

$$x_{0,0} \qquad (\mathit{Assertion}\ a)$$

   The resulting boolean formula is

$$(x_{0,0} \wedge p_{0,0} \wedge (x_{0,0} \wedge p_{0,0})) \wedge \neg x_{0,0}$$

   This formula is not satisfiable. Therefore, there is no counter-example for the given scope.

   **Scope 2:** For checking `check demo for 2` we encode the constraints as:

$$\begin{aligned} c &:= ((x_{0,0} \wedge \neg x_{0,1}) \vee (\neg x_{0,0} \wedge x_{0,1})) \wedge ((x_{1,0} \wedge \neg x_{1,1}) \vee (\neg x_{1,0} \wedge x_{1,1})) \\ c_1 &:= ((p_{0,0} \wedge \neg p_{0,1}) \vee (\neg p_{0,0} \wedge p_{0,1})) \wedge ((p_{1,0} \wedge \neg p_{1,1}) \vee (\neg p_{1,0} \wedge p_{1,1})) \\ c_2 &:= ((x_{0,0} \wedge p_{0,0}) \vee (x_{0,1} \wedge p_{1,0})) \wedge ((x_{1,0} \wedge p_{0,1}) \vee (x_{1,1} \wedge p_{1,1})) \end{aligned}$$

   and the assertion is encoded as before

$$a := (x_{0,0} \vee x_{1,0}) \wedge (x_{0,1} \vee x_{1,1})$$

   The resulting boolean formula is

$$c \wedge c_1 \wedge c_2 \wedge \neg a$$

This formula is not satisfiable. Therefore, there is no counter-example for the given scope.

**Larger Scopes:** From the new fact we conclude that no node is the next of two other nodes. Therefore, we will not find a counter-example to the assertion for larger scopes.