ETH zürich

P. Müller and D. Basin

Formal Methods and Functional Programming

Optional Exercises 10: States and Expressions

The solutions of the assignments can be found at the end of the file.

Assignment 4 (Similar States)

In this assignment, we show that an expression will always be evaluated the same way in two different states, provided that the states agree on the values assigned to the free variables of the expression.

Task: Prove that

$$\forall \sigma, \sigma', e \cdot \big(\big(\forall x \cdot x \in FV(e) \implies \sigma(x) = \sigma'(x)\big) \implies \mathcal{A}[\![e]\!]\sigma = \mathcal{A}[\![e]\!]\sigma'\big).$$

Assignment 5 (Substitutions on Expressions)

Substitutions on expressions do not necessarily commute; that is, it is not always the case that $e[x \mapsto e_1][y \mapsto e_2] \equiv e[y \mapsto e_2][x \mapsto e_1]$. In general, three extra conditions need to be imposed for substitutions to commute, namely $x \not\equiv y$, $x \notin FV(e_2)$, and $y \notin FV(e_1)$.

We see that the first condition is necessary: If $x \equiv y$ were allowed, we could choose $e \equiv x$, $e_1 \equiv 1$, and $e_2 \equiv 2$; then $e[x \mapsto e_1][y \mapsto e_2] \equiv 1 \not\equiv 2 \equiv e[y \mapsto e_2][x \mapsto e_1]$.

Task 5.1: Show that the second condition $x \notin FV(e_2)$ is necessary.

Note: Due to symmetry, this also shows that the third condition $y \notin FV(e_1)$ is necessary.

Task 5.2: Prove that substitution on expressions commutes if the three aforementioned conditions hold, i.e., prove that

$$\forall x, y, e_1, e_2 \cdot \big(x \not\equiv y \wedge y \notin FV(e_1) \wedge x \notin FV(e_2) \implies \\ \forall e \cdot (e[x \mapsto e_1][y \mapsto e_2] \equiv e[y \mapsto e_2][x \mapsto e_1])\big).$$

Hint: You may use the following lemma (which was proved in the exercise session):

$$\forall x, e, e' \cdot \big(x \notin FV(e) \implies (e[x \mapsto e'] \equiv e)\big)$$

Task 5.3: Consider the following closely-related statement (which is stated in terms of the interpretations of the two expressions):

$$\forall \sigma, x, y, e_1, e_2 \cdot \big(x \not\equiv y \wedge y \notin FV(e_1) \wedge x \notin FV(e_2) \implies \\ \forall e \cdot (\mathcal{A}[\![e[x \mapsto e_1][y \mapsto e_2]]\!]\sigma = \mathcal{A}[\![e[y \mapsto e_2][x \mapsto e_1]]\!]\sigma)\big)$$

A simple way to prove this is to just use the result from Task 4.2. Find an alternative proof, which does not rely on this result and which does not require a further induction argument.

Hint: Consider using other results mentioned on this exercise sheet.

Note: This exercise is a bit more involved.

Solutions: Assignment 4 (Similar States)

Let the states σ and σ' be arbitrary. We define

$$P(e) \equiv \big((\forall x \cdot x \in FV(e) \implies \sigma(x) = \sigma'(x)) \implies \mathcal{A}[\![e]\!]\sigma = \mathcal{A}[\![e]\!]\sigma'\big)$$

and prove $\forall e \cdot P(e)$ by strong structural induction on the arithmetic expression e. That is, we prove $P(e)$ for an arbitrary arithmetic expression e and assume $\forall e' \sqsubset e \cdot P(e')$ as our induction hypothesis.

In order to prove the implication, we assume that $\forall x \cdot x \in FV(e) \Rightarrow \sigma(x) = \sigma'(x)$, and seek to conclude that $\mathcal{A}[\![e]\!]\sigma = \mathcal{A}[\![e]\!]\sigma'$. We proceed by a case analysis on e:

- **Case** $e \equiv n$, for some numerical value n: Using the definition of $\mathcal{A}$, we get

  $$\mathcal{A}[\![n]\!]\sigma = \mathcal{N}[\![n]\!] = \mathcal{A}[\![n]\!]\sigma'.$$

- **Case** $e \equiv y$, for some variable y: Note that $FV(y) = \{y\}$. Thus, by our assumption, we have $\sigma(y) = \sigma'(y)$. Using this, we get

  $$\mathcal{A}[\![y]\!]\sigma = \sigma(y) = \sigma'(y) = \mathcal{A}[\![y]\!]\sigma',$$

  as required.

- **Case** $e \equiv e_1\, op\, e_2$, for some arithmetic expressions e_1, e_2 and some arithmetic operator op: Note that, for $i \in \{1, 2\}$, we have $e_i \sqsubset e$. Thus, by our induction hypothesis, we have $P(e_i)$, i.e.,

  $$(\forall x \cdot x \in FV(e_i) \implies \sigma(x) = \sigma'(x)) \implies \mathcal{A}[\![e_i]\!]\sigma = \mathcal{A}[\![e_i]\!]\sigma'$$

  By definition of $FV(\cdot)$, we have $FV(e_1\, op\, e_2) = FV(e_1) \cup FV(e_2) \supseteq FV(e_i)$. Therefore, from our assumption, we can deduce that $\forall x \cdot x \in FV(e_i) \Rightarrow \sigma(x) = \sigma'(x)$ holds. Using this along with our induction hypothesis, we obtain that $\mathcal{A}[\![e_i]\!]\sigma = \mathcal{A}[\![e_i]\!]\sigma'$. Then, using this fact, we can obtain our desired conclusion as follows:

  $$\begin{aligned} \mathcal{A}[\![e_1\, op\, e_2]\!]\sigma &= \mathcal{A}[\![e_1]\!]\sigma\ \overline{op}\ \mathcal{A}[\![e_2]\!]\sigma && (\mathcal{A}) \\ &= \mathcal{A}[\![e_1]\!]\sigma'\ \overline{op}\ \mathcal{A}[\![e_2]\!]\sigma' && (\text{fact}) \\ &= \mathcal{A}[\![e_1\, op\, e_2]\!]\sigma' && (\mathcal{A}) \end{aligned}$$

Solutions: Assignment 5 (Substitutions on Expressions)

Task 5.1: If $x \in FV(e_2)$ were allowed (but $x \not\equiv y$ is still required), we could choose $e \equiv y$, $e_1 = 1$ and $e_2 \equiv x$; then

$$e[x \mapsto e_1][y \mapsto e_2] \equiv y[x \mapsto 1][y \mapsto x] \equiv x \not\equiv 1 \equiv y[y \mapsto x][x \mapsto 1] \equiv y[y \mapsto e_2][x \mapsto e_1].$$

Task 5.2: Let the variables x, y, and the expressions e_1, e_2 be arbitrary. We show the implication, by assuming its left-hand side, i.e., $x \not\equiv y$, $y \notin FV(e_1)$ and $x \notin FV(e_2)$, and aim to prove its right-hand side. To this end, we define

$$P(e) \equiv \big(e[x \mapsto e_1][y \mapsto e_2] \equiv x[y \mapsto e_2][x \mapsto e_1]\big)$$

and prove $\forall e \cdot P(e)$ by structural induction on e:

- **Numeral Case:** We need to prove $P(n)$, for some numerical value n. By the definition of substitution, we have

$$n[x \mapsto e_1][y \mapsto e_2] \equiv n[y \mapsto e_2] \equiv n \equiv n[x \mapsto e_1] \equiv n[y \mapsto e_2][x \mapsto e_1].$$

- **Variable Case:** We need to prove $P(z)$ for some variable z. We distinguish the following cases:
  - **Case** $z \equiv x$: Note that, by our assumption, we have $z \not\equiv y$. By the definition of substitution, we get

$$\begin{aligned} z[z \mapsto e_1][y \mapsto e_2] &\equiv e_1[y \mapsto e_2] \\ &\equiv e_1 && \text{(L)} \\ &\equiv z[z \mapsto e_1] \\ &\equiv z[y \mapsto e_2][z \mapsto e_1], && (z \not\equiv y) \end{aligned}$$

    where the step marked with (L) follows by the lemma on the exercise sheet together with the assumption $y \notin FV(e_1)$.
  - **Case** $z \equiv y$: By symmetric argument to the previous case.
  - **Case** $z \not\equiv x$ and $z \not\equiv y$: By the definition of substitution, we have

$$z[x \mapsto e_1][y \mapsto e_2] \equiv z[y \mapsto e_2] \equiv z \equiv z[x \mapsto e_1] \equiv z[y \mapsto e_2][x \mapsto e_1],$$

    as required.
- **Operation Case:** We need to prove $P(e_3\ op\ e_4)$, for some arithmetic expressions e_3, e_4, and some arithmetic operator op. Our induction hypothesis is $P(e_1)$ and $P(e_2)$. Combining this with the definition of substitution yields

$$\begin{aligned} (e_3\ op\ e_4)[x \mapsto e_1][y \mapsto e_2] &\equiv (e_3[x \mapsto e_1]\ op\ e_4[x \mapsto e_1])[y \mapsto e_2] \\ &\equiv e_3[x \mapsto e_1][y \mapsto e_2]\ op\ e_4[x \mapsto e_1][y \mapsto e_2] \\ &\equiv e_3[y \mapsto e_2][x \mapsto e_1]\ op\ e_4[y \mapsto e_2][x \mapsto e_1] && \text{(IH)} \\ &\equiv (e_3[y \mapsto e_2]\ op\ e_4[y \mapsto e_2])[x \mapsto e_1] \\ &\equiv (e_3\ op\ e_4)[y \mapsto e_2][x \mapsto e_1]. \end{aligned}$$

Task 5.3: Let the state σ, the variables x, y, and the arithmetic expressions e_1, e_2 be arbitrary. We aim to prove the right-hand side of an implication, so we assume its left-hand side, i.e. $x \not\equiv y$, $y \notin FV(e_1)$ and $x \notin FV(e_2)$. For some arbitrary arithmetic expression e, we have

$$\begin{aligned}
\mathcal{A}[\![e[x \mapsto e_1][y \mapsto e_2]]\!]\sigma &= \mathcal{A}[\![e[x \mapsto e_1]]\!](\sigma[y \mapsto \mathcal{A}[\![e_2]\!]\sigma]) && (**)\\
&= \mathcal{A}[\![e]\!]\big(\sigma[y \mapsto \mathcal{A}[\![e_2]\!]\sigma][x \mapsto \mathcal{A}[\![e_1]\!](\sigma[y \mapsto \mathcal{A}[\![e_2]\!]\sigma])]\big) && (**)\\
&= \mathcal{A}[\![e]\!]\big(\sigma[y \mapsto \mathcal{A}[\![e_2]\!]\sigma][x \mapsto \mathcal{A}[\![e_1]\!]\sigma]\big) && \text{(A3)}\\
&= \mathcal{A}[\![e]\!]\big(\sigma[x \mapsto \mathcal{A}[\![e_1]\!]\sigma][y \mapsto \mathcal{A}[\![e_2]\!]\sigma]\big) && \text{(Task 1.2)}\\
&= \mathcal{A}[\![e]\!]\big(\sigma[x \mapsto \mathcal{A}[\![e_1]\!]\sigma][y \mapsto \mathcal{A}[\![e_2]\!](\sigma[x \mapsto \mathcal{A}[\![e_1]\!]\sigma])]\big) && \text{(A3)}\\
&= \mathcal{A}[\![e[y \mapsto e_2]]\!](\sigma[x \mapsto \mathcal{A}[\![e_1]\!]\sigma]) && (**)\\
&= \mathcal{A}[\![e[y \mapsto e_2][x \mapsto e_1]]\!]\sigma, && (**)
\end{aligned}$$

where the equalities marked with (A3) follow by Assignment 3 with our assumptions $y \notin FV(e_1)$ and $x \notin FV(e_2)$, respectively. Note that we were only allowed to apply the result from Task 1.2 since $x \not\equiv y$.