ETH zürich

P. Müller and C. Sprenger

Formal Methods and Functional Programming

Exercise Sheet 10: States and Expressions

Submission deadline: May 9/10, 2023

Submit by May 4th 23:59 to receive feedback via email before the midterm on May 9th. Thus, we will also accept submissions during the May 2/3 sessions or (exceptionally) via email.

Assignment 1 (Simplifying State Updates)

In this assignment, we will prove statements that allow us to "clean up" states as we apply additional state updates to them: If many state updates to the same variable have been applied, you can always leave out all but the last one (Tasks 1.1 and 1.3), and reordering state updates applied to different variables never change the state (Task 1.2).

Task 1.1: Prove that, for all states σ, all variables x, and all values v_1 and v_2, we have

$$\sigma[x \mapsto v_1][x \mapsto v_2] = \sigma[x \mapsto v_2].$$

Task 1.2: Prove that, for all states σ, all variables x and y with $x \not\equiv y$, and all values v and w, we have

$$\sigma[x \mapsto v][y \mapsto w] = \sigma[y \mapsto w][x \mapsto v].$$

Task 1.3: Prove that, for all states σ, all variables x, all values v_1 and v_2, all natural numbers n, and all sequences $\vec{y} \equiv \langle y_1, \ldots, y_n \rangle$ and $\vec{w} \equiv \langle w_1, \ldots, w_n \rangle$ of variables and values, respectively, we have

$$\sigma[x \mapsto v_1][\vec{y} \mapsto \vec{w}][x \mapsto v_2] = \sigma[\vec{y} \mapsto \vec{w}][x \mapsto v_2],$$

where $\sigma[\vec{y} \mapsto \vec{w}]$ denotes the sequence of updates $\sigma[y_1 \mapsto w_1] \ldots [y_n \mapsto w_n]$.

Note: This task is a bit more involved.

Assignment 2 (Substitution Properties)

In the exercise session, we proved a substitution lemma for arithmetic expressions:

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

Task: Prove the corresponding statement for boolean expressions, i.e, prove that

$$\forall \sigma, b, e, x \cdot \big(\mathcal{B}[\![b[x \mapsto e]]\!]\sigma = \mathcal{B}[\![b]\!](\sigma[x \mapsto \mathcal{A}[\![e]\!]\sigma])\big).$$

Assignment 3 (Applying Big-Step Semantics)

Consider the following statement s:

```
while n # 0 do
    (a := a + n;
     b := b * n);
     n := n - 1
end
```

Task. Let σ be a state such that $\sigma(\mathtt{a}) = 0$, $\sigma(\mathtt{b}) = 1$, and $\sigma(\mathtt{n}) = 2$. Prove using the natural semantics that there is some state σ' with $\sigma'(\mathtt{a}) = 3$, $\sigma'(\mathtt{b}) = 2$, and $\sigma'(\mathtt{n}) = 0$ such that $\langle s, \sigma\rangle \to \sigma'$.

Provide the complete derivation tree. Do not forget to explicitly write the names of the rules you apply at each derivation step.