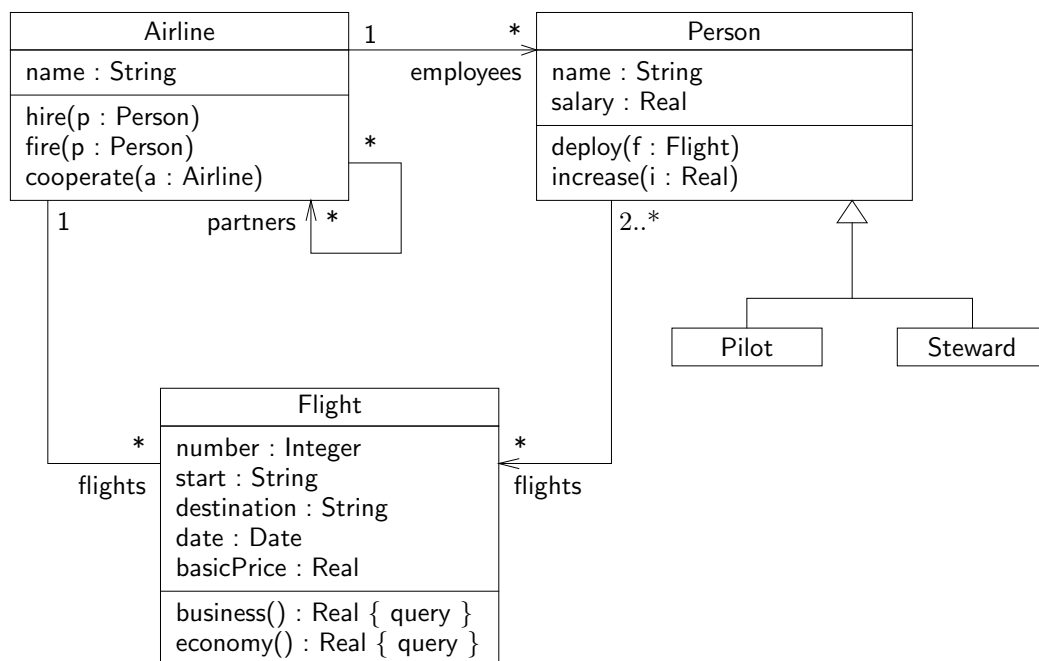


Übungen zu Formale objektorientierte Software-Entwicklung

Aufgabe 1

Gegeben sei das Klassendiagramm für den Anwendungsbereich „Fluglinien“.



Seien $a : \text{Airline}$, $p : \text{Person}$, $r : \text{Real}$, $ab : \text{Bag}(\text{Airline})$, $ps : \text{Set}(\text{Pilot})$ getypte Variablen. Entscheiden Sie, ob die im Folgenden aufgeführten Zeichenketten korrekte OCL-Ausdrücke über diesem Klassendiagramm und den angegebenen Variablen sind. Geben Sie für die korrekten OCL-Ausdrücke den induktiven Aufbau des Ausdrucks anhand der Regeln (1–8) aus Abschnitt 2.3 der Vorlesung an.

1. $a.\text{partners} \rightarrow \text{iterate}(a : \text{Airline}; ab : \text{Bag}(\text{Airline}) = \text{Bag}\{\} \mid ab.\text{including}(a))$
2. $a.\text{employees} \rightarrow \text{size}() > 0$ and
 $a.\text{employees} \rightarrow \text{collect}(p : \text{Person} \mid p.\text{salary}) \rightarrow \text{sum}() / a.\text{employees} \rightarrow \text{size}()$
3. $a.\text{employees} \rightarrow \text{exists}(p : \text{Person} \mid p = \text{null})$

4. `Person.allInstances@pre()->select(p : Pilot | p.oclIsNew())`
5. `if a.partners->size() > 0 then a.partners else a.partners->asSequence() endif`
6. `a.employees->iterate(p : Person; ps : Set(Pilot) = Set{} |
if p.oclIsTypeOf(Pilot) then ps->including(p) else ps endif)`
7. `a.employees@pre->iterate(p : Person; r : Real = 0.0 | p.salary@pre+r)`

Aufgabe 2

Zeigen Sie, dass gilt:

1. Für alle $x, y \in \llbracket \text{Boolean} \rrbracket$ ist

$$\llbracket \text{implies} \rrbracket(x, y) = \llbracket \text{or} \rrbracket(\llbracket \text{not} \rrbracket(x), \llbracket \text{and} \rrbracket(x, y))$$

2. Für alle $x \in \llbracket \text{Collection}(\text{OclAny}) \rrbracket$ und $y \in \llbracket \text{OclAny} \rrbracket$ ist

$$\llbracket \text{not} \rrbracket(\llbracket \text{->includes}(_) \rrbracket(x, y)) = \llbracket \text{->excludes}(_) \rrbracket(x, y)$$

Aufgabe 3

Sei Δ ein Klassendiagramm.

Definieren Sie eine geeignete Familie von semantischen Funktionen

$$\llbracket \text{->first}(_) \rrbracket : \llbracket \text{Sequence}(T) \rrbracket \rightarrow \llbracket T \rrbracket$$

für $T \in \text{Base}_{\Delta}^{\text{OCL}} \cup \text{Class}_{\Delta}^{\text{OCL}}$, so dass $\llbracket \text{->first}(_) \rrbracket(x)$ das erste Element einer Sequenz $x \in \llbracket \text{Sequence}(T) \rrbracket$ berechnet.

Aufgabe 4

Sei Δ ein Klassendiagramm und \leq die partielle Ordnung der über Δ definierten sorten-geordneten OCL-Signatur $\Sigma_{\Delta}^{\text{OCL}} = (S_{\Delta}^{\text{OCL}}, \leq, OP_{\Delta}^{\text{OCL}})$.

Zeigen Sie, dass für alle Sortensymbole $T, T' \in S_{\Delta}^{\text{OCL}}$ gilt: Ist $T \leq T'$, so gilt $\llbracket T \rrbracket \subseteq \llbracket T' \rrbracket$.