

1 FOOSE SoSe 2010, Lösungsvorschlag zu Übungsblatt 2

1.1 Aufgabe 1

Gegeben sei das auf Übungsblatt 1 angegebene Klassendiagramm für den Anwendungsbe-
reich „Fluglinien“.

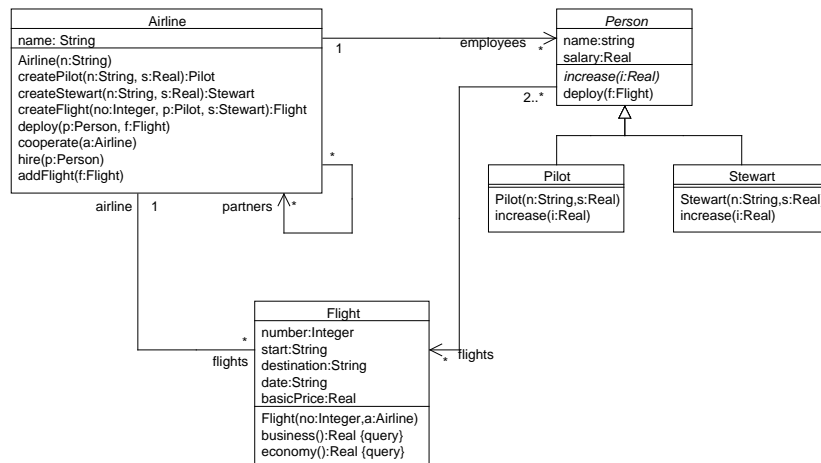


Abbildung 1: Klassendiagramm

Seien $self : Airline$, $a : Airline$, $as : Set(Airline)$, $p : Person$, $ps : Set(Person)$, $p : Pilot$, $f : Flight$, $a : Flight$, $r : Integer$ getypte Variablen. Entscheiden Sie, welche der im Folgenden aufgeführten Zeichenketten korrekte OCL-Ausdrücke über dem Klassendiagramm und den angegebenen Variablen sind. Geben Sie für jeden korrekten OCL-Ausdruck den induktiven Aufbau in Form einer Baumdarstellung an sowie den Typ des Ausdrucks und die darin vorkommenden freien Variablen.

1. `as.includes(a)`

Lösung: Nicht korrekt. Richtig wäre `as->includes(a)`.

2. `self.partners->iterate(a : Airline; as : Set(Airline) = Set{} | as->including(a))`

Lösung: Korrekt.

Typ: `Set(Airline)`.

$FV = \{self : Airline\}$

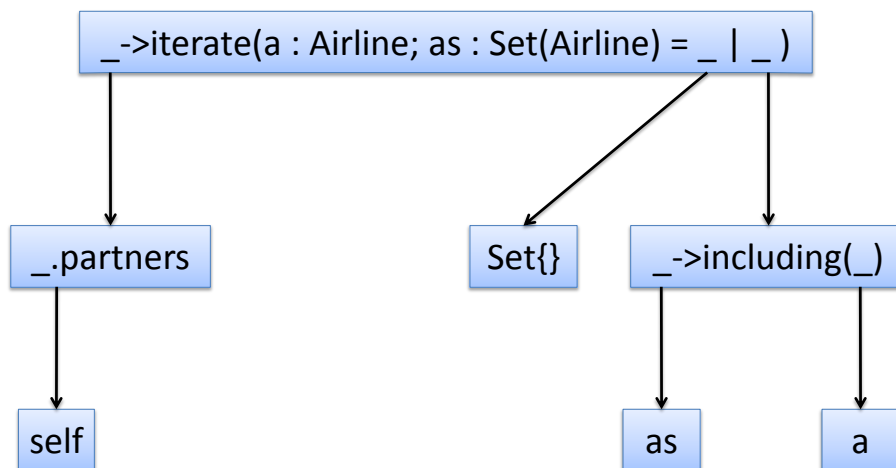


Abbildung 2: Baumdarstellung zu 2.

3. `a.employees->size() > 0 and a.employees->any(p : Person | p.salary > 5000)`

Lösung: Nicht korrekt. Der Any-Teilausdruck hat den Typ `Person` statt `Boolean`.

4. `a.employees->exists(p | p = null)`

Lösung: Korrekt.

Typ: `Boolean`

$FV = \{a : Airline\}$

5. `Person.allInstances@pre()->select(p : Pilot | p.oclIsNew())`

Lösung: Nicht korrekt. `p` muss vom Typ `Person` sein.

6. `if a.partners->size() > 0 then a.partners else a.partners->asSequence() endif`

Lösung: Korrekt.

Typ: `Collection(Airline)`

$FV = \{a : \text{Airline}\}$

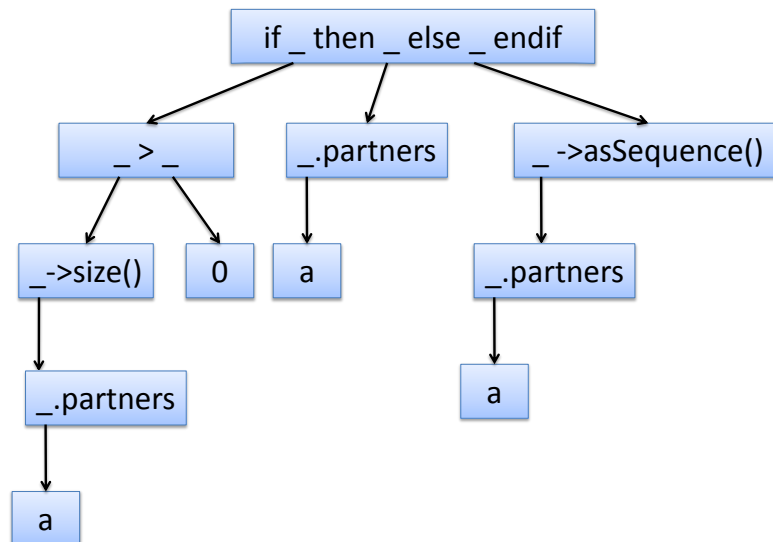


Abbildung 3: Baumdarstellung zu 6.

7. `a.employees->iterate(p : Person; ps : Set(Person) = Set{} |
 if p.oclIsTypeOf(Pilot) then ps->including(p) else ps endif)`

Lösung: Korrekt.
 Typ: Set(Person)
 FV = {a : Airline}

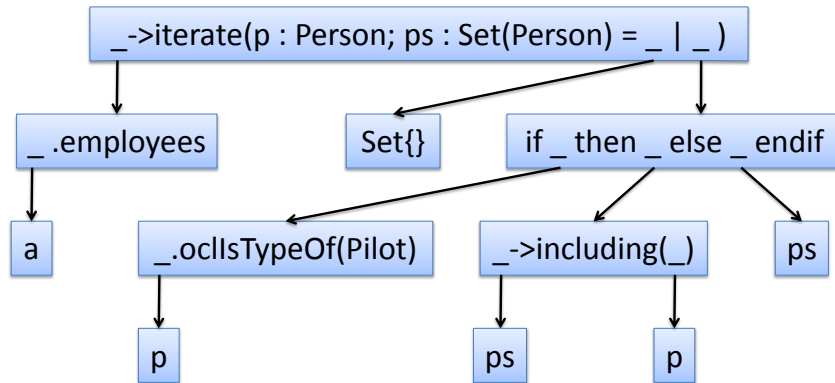


Abbildung 4: Baumdarstellung zu 7.

8. `a.employees@pre->iterate(p : Person; r : Integer = 0 | r+p.salary@pre)`

Lösung: Nicht korrekt. Das Attribut `salary` der Klasse `Person` hat den Typ `Real`. Dadurch hat der Teilausdruck `r+p.salary@pre` den Typ `Real`. Der Variablen `r : Integer` kann jedoch kein `Real`-Wert zugewiesen werden.

9. `a.flights->forAll(f : Flight | f.basicPrice + 100)`

Lösung: Nicht korrekt. Der Teilausdruck `f.basicPrice + 100` hat den Type `Real`, müsste jedoch den Type `Boolean` haben.

10. `a.flights->forAll(a : Flight | a.basicPrice > 100)`

Lösung: Korrekt.

Typ: `Boolean`

$FV = \{a : \text{Airline}\}$

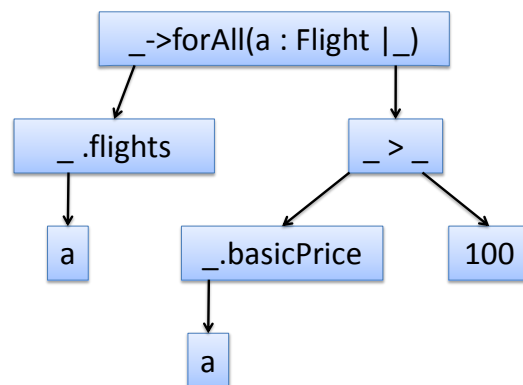


Abbildung 5: Baumdarstellung zu 10.