

## Übungen zu Formale Objektorientierte Software-Entwicklung

### Aufgabe 1

Gegeben sei das auf Übungsblatt 1 angegebene Klassendiagramm für den Anwendungsbereich „Fluglinien“.

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)`
2. `self.partners->iterate(a : Airline; as : Set(Airline) = Set{} | as->including(a))`
3. `a.employees->size() > 0 and a.employees->any(p : Person | p.salary > 5000)`
4. `a.employees->exists(p | p = null)`
5. `Person.allInstances@pre()->select(p : Pilot | p.oclIsNew())`
6. `if a.partners->size() > 0 then a.partners else a.partners->asSequence() endif`
7. `a.employees->iterate(p : Person; ps : Set(Person) = Set{} | if p.oclIsTypeOf(Pilot) then ps->including(p) else ps endif)`
8. `a.employees@pre->iterate(p : Person; r : Integer = 0 | r+p.salary@pre)`
9. `a.flights->forAll(f : Flight | f.basicPrice + 100)`
10. `a.flights->forAll(a : Flight | a.basicPrice > 100)`