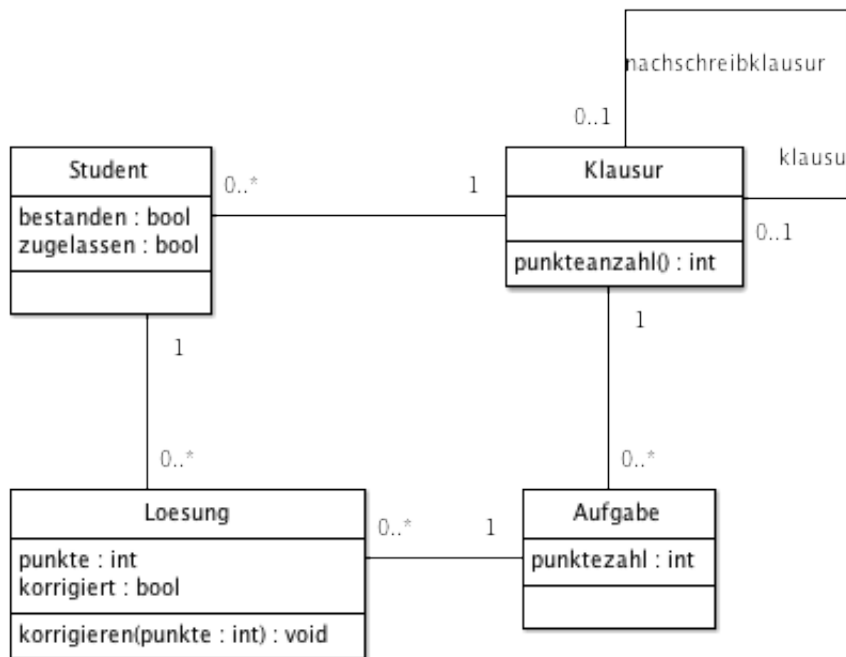


2 OCL

1.
 - a) Die Klasse Lösung repräsentiert nach meinem Verständnis die Lösung mit den damit eingebrachten Punkten für eine bestimmte Aufgabe, die ein Student gelöst hat.
 - b) Die Methode wird die in der Lösung erreichten Punkte der in der Klausur gestellten Aufgaben zusammenrechnen.
 - c)
2.
 - a) Für jede Aufgabe gibt es mindestens einen Punkt.
 - b) Jeder Student kann nur genauso viele Lösungen abgeben, wie es Aufgaben in der Klausur gibt.
 - c) Für alle Studenten, die die Klausur bestanden haben, haben bei einer Aufgabe mehr als null Punkte erreicht.
3.
 - a) `context Klausur inv: aufgaben->exists(a | a.punktzahl = 1)`
 - b) `context Klausur inv:`
`nachschreibklausur->exists(n | not n.nachschreibeklausur->exists())`
 - c) `context Student inv: zugelassen = true implies`
`klausur.aufgaben->forall (a | a.loesungen->`
`exists(l | l.student = self and l.aufgabe = a))`
4.
 - a) Die Methode korrigieren() soll aus der Korrektur des Professors die erreichten Punkte der Lösung berechnen und diese in punkte speichern. Dazu werden die Punkte benötigt, die erreicht worden sind. Die Methode setzt auch die korrigiert-Variable auf true, damit man

Lösungen mit erreichten Punkten von noch nicht korrigierten Lösungen unterscheiden kann.



b) context `Loesung::korrigieren(p)` pre:

`korrigiert = false and punkte = 0`

c) context `Loesung::korrigieren(p)` post:

`korrigiert = true and punkte = p`

3 Tolle Trolle in OCL

context `Troll::istToll():boolean` post:

`if keulen->size() > 1 then result=true`

`else result= false endif`