

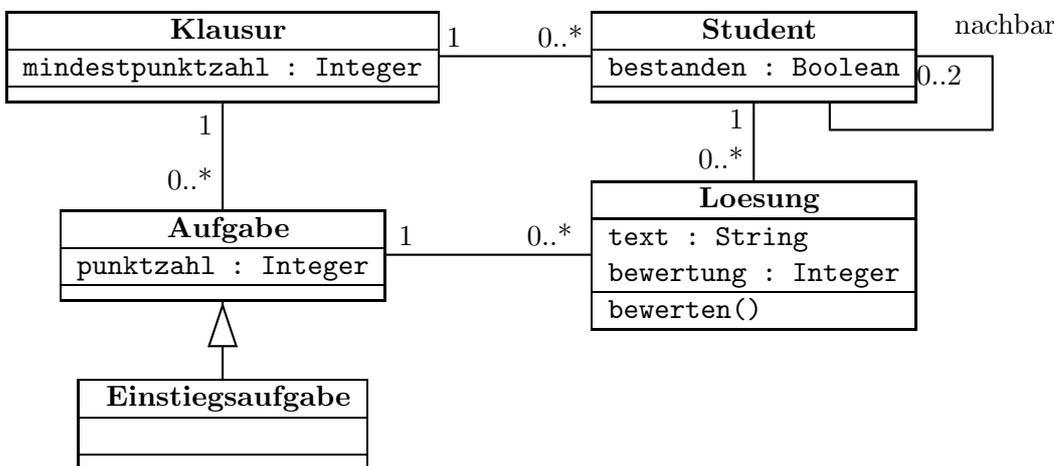
## Praktikum

### Formale Entwicklung objektorientierter Software

#### Übungsblatt 4

#### Aufgabe 1

Das folgende UML-Klassendiagramm sei gegeben:



Betrachten Sie das Java-Programm

<http://i12www.ira.uka.de/~engelc/lehre/keypraktWS0708/Blatt4.java>

Es implementiert die Klassen des obigen Klassendiagramms.

- (a) Übersetzen Sie folgende natürlichsprachliche Beschreibungen in JML-Spezifikationen (Invarianten, Methodenkontrakte) und fügen Sie diese in das obige Java-Programm ein:
- Für jede Aufgabe gilt, dass ihre Punktzahl größer als Null ist.
  - Die Zahl der Lösungen eines Studenten ist gleich der Zahl der Aufgaben seiner Klausur. Drücken Sie diesen Sachverhalt als Kontrakt für die Methode `Student.macheLoesungen(Loesung[] loesungen)` aus.
  - Die folgende Bedingung soll sich ebenfalls in einem Kontrakt für diese Methode wiederfinden: Ein Student hat nicht bestanden, wenn er eine Lösung hat, deren Text mit dem Text einer Lösung eines seiner Nachbarn übereinstimmt.
  - Für jede Einstiegsaufgabe gilt: Ihre Punktzahl ist kleiner oder gleich der Punktzahl aller Aufgaben der Klausur, zu der sie gehört.
  - Nachdem eine Lösung bewertet worden ist (d.h., nach Ausführung von `bewerten()`), ist ihre Bewertung größer oder gleich Null und kleiner oder gleich der Punktzahl der Aufgabe zu der sie gehört.

vi. Verwenden Sie zur Modellierung des folgenden Sachverhalts das `\sum`-Konstrukt

Die Summe der Punktzahlen aller Aufgaben einer Klausur ist größer als die Mindestpunktzahl der Klausur.

- (b) Erfüllt die Implementierung auch die in den vorherigen Teilaufgaben erstellten Spezifikationen?

Wenn ja, (wie) könnten Sie das nachweisen? Wenn nein, implementieren Sie die `main`-Methode so, dass nach ihrer Ausführung ein Systemzustand eintritt, der eine der spezifizierten Invarianten nicht erfüllt und geben Sie an, welche Invariante verletzt wird.

## Aufgabe 2

Laden Sie folgende Datei herunter und entpacken Sie diese:

<http://i12www.ira.uka.de/~engelc/lehre/keypraktWS0708/core.tgz>

Im Ordner `core` finden Sie nun unter anderem die Datei `ExamDataBase.java` und die dazugehörige Dokumentation im `html`-Format. Benutzen Sie diese Dokumentation, um folgende Eigenschaften in `JML` zu spezifizieren:

- (a) die Invariante für die Klasse `ExamDataBase`  
(b) den Vertrag für die Methode `setPoints` in der Klasse `ExamDataBase`

### Abgabe bis 21.11.

Es braucht pro Gruppe nur *eine* Lösung abgegeben werden.

Die Abgabe der Übungsblätter erfolgt mit dem `SVN` System. Dazu legen Sie die abzugebenden Dateien im `SVN` ab und kopieren sie mit `SVN` in den

Unterordner `abgabe/<nr>` wie in Aufgabe 2 auf Blatt 1 beschrieben.

Einige Aufgaben verlangen eine schriftliche Bearbeitung, diese ist dann je nach Komplexität als `ASCII`, `html`, `ps`- oder `pdf`-Dokument abzugeben. Auf *keinen* Fall im `MS Word doc`-Format.

---

**Praktikums-Webseite:** <http://i12www.ira.uka.de/~engelc/lehre/keypraktWS0708/>

*Dr. Thomas Käufel:* Zi. 207, Tel. 608-6286, E-Mail: [kaeufl@ira.uka.de](mailto:kaeufl@ira.uka.de)

*Christian Engel:* Zi. 106, Tel. 608-4338, E-Mail: [engelc@ira.uka.de](mailto:engelc@ira.uka.de)

*Benjamin Weiß:* Zi. 309, Tel. 608-4324, E-Mail: [bweiss@ira.uka.de](mailto:bweiss@ira.uka.de)