



## **Praktikum**

### **Formale Entwicklung objektorientierter Software**

#### **Übungsblatt 8**

### **Praktikumsprojekt**

Als Inhaber einer Softwarefirma werden Sie vom Institut für Terraforming-Integrität (ITI) beauftragt, ein Softwaresystem zur Verwaltung von Klausurergebnissen zu entwickeln. Das ITI stellt an die Korrektheit dieses Systems hohe Anforderungen, die nur durch den konsequenten Einsatz formaler Methoden erfüllt werden können.

Das System soll den Mitarbeitern des ITI die Möglichkeit bieten, Datensätze über Klausurteilnehmer zu speichern und auszuwerten. Jeder Datensatz soll dabei aus Nachname, Vorname, Matrikelnummer, Punktzahl und Bonuspunktzahl (aus der Teilnahme an den vorlesungsbegleitenden Übungen) bestehen. Außerdem soll vermerkt werden, ob der Teilnehmer seine Anmeldung wieder zurückgezogen hat. Die Eingabe und Verwaltung der Datensätze soll über eine grafische Benutzeroberfläche erfolgen. Datensätze sollen schon angelegt und gespeichert werden können, bevor die Klausur geschrieben wurde.

Nach der Klausur soll es dann möglich sein, die Punktzahlen der Teilnehmer nachzutragen. Außerdem sollen die Bewertungsparameter der Klausur eingestellt werden können: Die Mindestpunktzahl zum Bestehen (Note 4, 0), die Maximalpunktzahl, und die Schrittweite zum Erreichen der jeweils nächstbesseren Notenstufe. Die Noten der Teilnehmer sollen dann abhängig von diesen Parametern automatisch errechnet werden. Beispielsweise soll sich bei einer Mindestpunktzahl von 20 Punkten und einer Schrittweite von 3 Punkten für einen Teilnehmer mit 27,5 Punkten die Note 3,3 ergeben.

Das System soll weiterhin in der Lage sein, die eingegebenen Daten in einer Datei zu speichern und wieder zu laden, und eine Übersicht im HTML-Format auszugeben. Außerdem soll eine grafische Darstellung der Notenverteilung als Balkendiagramm unterstützt werden.

Das ITI hat die Schnittstelle der zentralen Kernkomponente des zu entwickelnden Systems bereits in Form einer abstrakten Java-Klasse festgelegt und ihre Funktionalität mit JML formal spezifiziert. Man besteht darauf, dass Ihre Implementierung dieser Spezifikation exakt entspricht. Dies soll durch Verifikation mit dem KeY-System nachgewiesen werden.

## Aufgabe 1

Laden Sie von der Praktikumswebseite die Datei `core.tgz` herunter. Sie enthält die vorgegebene Schnittstelle der Kernkomponente des Klausurbewertungssystems in Form der Klassen `ExamDataBase`, `Student` und `ExamDataBaseException`. Implementieren Sie `ExamDataBase`, indem Sie eine konkrete Unterklasse schreiben, welche die Spezifikation erfüllt. Die drei vorgegebenen Klassen dürfen Sie dabei in keiner Weise verändern.

*Hinweise:*

- Beachten Sie, dass es halbe und viertel Punkte geben kann. Punktzahlen sollten daher als Integer mit dem Wert “*echte Punktzahl \* 100*” kodiert werden. Ähnliches gilt für Notenwerte.
- Da die Korrektheit Ihrer Implementierung von `ExamDataBase` später mit dem KeY-System bewiesen werden soll, sollten Sie schon bei der Implementierung auf gute Verifizierbarkeit Ihres Codes achten. Insbesondere sollten Sie die folgenden Java-Features *nicht* verwenden:
  - Gleitkommazahlen, d.h. die Typen `float` und `double`.
  - Klassen der Java-Klassenbibliothek, mit Ausnahme von `java.lang.String`, oder sonstige Fremdklassen.
  - Features, die in *Java 5* neu eingeführt wurden, wie Generics, Enumerations, oder verbesserte `for`-Schleifen.
  - Automatische Konvertierungen in Strings (wie z.B. bei `"Hello World"+7`).
  - Innere Klassen.
- Falls Sie Hilfsmethoden schreiben, spezifizieren Sie diese mit JML.
- Die Möglichkeit von Integer-Überläufen können Sie ignorieren. Dieser Aspekt wird aus Gründen der Einfachheit auch später bei der Verifikation nicht betrachtet werden.
- Versuchen Sie im Moment noch nicht, Ihren Code mit KeY zu verifizieren. Diese Aufgabe wird Inhalt des 10. (letzten) Übungsblatts sein.

### Abgabe bis 23.01.

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 Käußl:* 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)