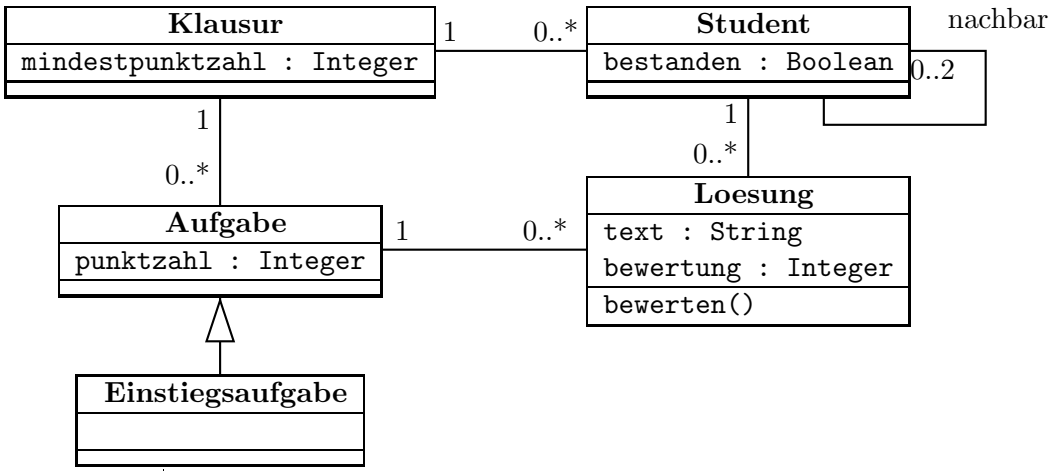
Universität Karlsruhe
Intitut für theoretische Informatik
Prof. Dr. P. H. Schmitt
C. Engel, F. Werner

7. November 2006

Praktikum
Formale Entwicklung objektorientierter Software
Übungsblatt 2

Aufgabe 4

Das folgende UML-Klassendiagramm sei gegeben:



Betrachten Sie das Java-Programm
`http://i12www.ira.uka.de/~engelc/lehre/keypraktWS0607/Blatt2.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:

i. Für jede Aufgabe gilt, dass ihre Punktzahl größer als Null ist.

ii. 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.

iii. 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.

iv. Für jede Einstiegsaufgabe gilt: Ihre Punktzahl ist kleiner oder gleich der Punktzahl aller Aufgaben der Klausur, zu der sie gehört.

v. 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 5

Als Inhaber der Firma "Mit SicherheIT" erhalten Sie den Auftrag, ein Authentifizierungssystem zu entwerfen und zu implementieren. Das System soll folgende Bedingungen erfüllen:

> Zur Authentifikation soll eine Chipkarte verwendet werden, auf der eine PIN (Personal Identification Number) gespeichert ist. Nachdem die Karte in den Kartenleser eingeführt wurde, hat der Benutzer drei Versuche, die richtige PIN einzugeben. Im Erfolgsfall gilt er als authentifiziert, im Negativfall wird die Karte als gesperrt vermerkt.

Spezifizieren Sie die Anforderungen an das Eingeben der PIN in JML (Invarianten bzw. Vor- und Nachbedingungen) und implementieren Sie ein entsprechendes System in Java. Diese Spezifikation soll so "vollständig" sein, dass es nicht möglich ist, die Operation, die das Eingeben der PIN modelliert, anders als oben beschrieben zu implementieren.

Erstellen Sie auch eine graphische Benutzeroberfläche für Ihr Programm unter Verwendung der Java Swing Bibliothek. Wird die Karte als gesperrt vermerkt, so ist das Eingabefeld zu deaktivieren und die Nachricht `Zugriff verweigert` anzuzeigen. Im Erfolgsfall soll dem Benutzer die Meldung `Zugriff gestattet` angezeigt werden. Das Programm soll zu jeder Zeit über einen Knopf mit der Auffschrift "Verlassen" beendet werden können.

Achten Sie auf eine strikte Trennung von Oberfläche und eigentlicher Programmlogik; legen Sie die GUI und und die das JML-Modell implementierenden Klassen, in getrennten Java `packages` ab, den Modellklassen ist es *verboten*, eine der GUI Klassen zu kennen.

Abgabe bis 22.11.
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 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/keypraktWS0607/`
Christian Engel: Email: `engelc@ira.uka.de`
Frank Werner: Zi. 308, Tel. 608-7322, Email: `werner@iti.uni-karlsruhe.de`