

Übungen zur Vorlesung Softwaretechnologie

- Wintersemester 2010/2011 -

Dr. Günter Kniesel

Übungsblatt 10

Zu bearbeiten bis: 9.01.2011

Bitte fangen Sie **frühzeitig** mit der Bearbeitung an, damit wir Ihnen bei Bedarf helfen können. Checken Sie die Lösungen zu den Aufgaben in Ihr SVN-Repository ein, Diagramme als VP-Dateien, Texte als Textdatei. Fragen zu Übungsaufgaben/Vorlesung können Sie auf der Mailingliste swt-tutoren@lists.iai.uni-bonn.de, bzw. swt-vorlesung@lists.iai.uni-bonn.de stellen.

Aufgabe 1. CRC-Karten (7 Punkte)

Es geht um die Modellierung der folgenden Bibliotheks-Anwendung:

Die Unibibliothek verwaltet Bücher, Zeitschriften und Seminararbeiten. Diese Gegenstände sollen rechnergestützt gefunden und ausgeliehen werden.

Zur Identifikation eines Benutzers bei Ausleihe und Rückgabe dient der Informatik-Login. Jeder Benutzer kann höchstens sieben Medien entleihen. Die Ausleihfrist beträgt bei Büchern 4 Wochen, bei Zeitschriften 2 Wochen und bei Seminararbeiten 1 Woche. Die Überziehungsgebühr beträgt bei Büchern 1 €, bei Zeitschriften 3 €, bei Seminararbeiten 5 € pro Tag der Überschreitung.

Benutzer dürfen nur dann weitere Medien ausleihen, wenn sie noch keine 42€ Überziehungsgebühren angesammelt haben und auch bei keinem entliehenen Medium die Frist überziehen.

- a) Erstellen Sie ein minimales Objektmodell für die Bibliotheksanwendung:
 - a. Eine textuelle Aufzählung von Klassen mit aussagekräftigen Namen.
 - b. Je ein kurzer prägnanter Text (je < 50 Wörter) der die Aufgabe der Klasse beschreibt.
- b) Unter Einsatz von CRC-Cards¹ soll das Objektmodell um Verantwortlichkeiten und Kollaborationen erweitert werden, und zwar so, dass die Klassen anschließend alles beinhalten, was für folgende Szenarien erforderlich ist:
 - a. Suchen
 - b. Erfolgreiches Ausleihen
 - c. Verweigertes Ausleihen
 - d. Rückgabe ohne Überziehung
 - e. Rückgabe mit Überziehung

Wenden Sie die in der Vorlesung eingeführten Techniken an. Es kann durchaus sein, dass Sie auf neue sinnvolle Klassen stoßen. Beschreiben Sie diese gegebenenfalls auch durch je eine CRC-Karte.

¹ Abgabe in Tabellenform als PDF-Dokument

Aufgabe 2. *Design by Contract* (7 Punkte)

Verfeinern Sie ihren Entwurf zu Aufgabe 1 wie folgt:

- Erstellen Sie ein Klassendiagramm, das die Ergebnisse der CRC-Analyse widerspiegelt, aber komplett typisiert ist (Ergebnis- und Parametertypen zu jeder Operation). Achten Sie darauf, dass manche Beziehungen evtl. eigene Attribute und Operationen besitzen.
- Beschreiben Sie für drei Operationen mit unterschiedlichen Verantwortlichkeiten, jeweils ihre Vor- und Nachbedingungen, die sich aus Aufgabe 1 ableiten lassen. Versuchen sie diese so genau wie möglich zu formulieren, unter Verwendung der Object Constraint Language (OCL).

Aufgabe 3. *Jahreszeitbedingte Anwendung von Entwurfsmustern* (16 Punkte)

Ein reichlich geschmückter Weihnachtsbaum besteht bekannterweise ausschließlich aus *Baumbestandteilen*, d.h. aus *Zweigen*² (an denen weitere Zweige wachsen können) und am Ende eines Zweiges evtl. *Baumschmuck*. Baumschmuck können *Engel*, *Kugeln*, *Sterne*, oder *Kerzen* sein.



- Modellieren Sie den Weihnachtsbaum in Java unter Zuhilfenahme eines geeigneten Entwurfsmusters. Markieren Sie dabei die Rollen des Entwurfsmusters in dem zur entsprechenden Klasse bzw. Methode gehörenden JavaDoc-Kommentar.
- Entwickeln Sie ein Programm, das Ausgehend vom Stamm einen zufällig generierten Weihnachtsbaum mit verschiedenem Baumschmuck erzeugt. Achten Sie dabei darauf, dass in unserem Kulturkreis die maximale Zweigrekursionstiefe 3 beträgt und jeder Zweig maximal 5 Kindelemente haben kann. Geben Sie eine Beschreibung ihres Baumes auf der Konsole aus, indem Sie die *toString()*-Methoden der Baumelemente geeignet implementieren und auf dem Stamm aufrufen.
- [Zur Baumpflege werden im Rheinland traditionell Heinzelmännchen eingesetzt](#), kleine Spezialisten, die den Baum entlang klettern und eine spezifische Aktion auf den konkreten Baumelementen ausführen können. Modellieren Sie in Ihrem Projekt unter Nutzung eines geeigneten Entwurfsmusters ein abstraktes *Heinzelmännchen*. Ändern Sie, wenn nötig auch bestehende Klassen, so dass später beliebige konkrete Heinzelmännchen die Baumpflege übernehmen können. Beachten Sie, dass die Ausprägung einer konkreten Heinzelmännchen-Aktion je nach Baumelement unterschiedlich sein kann! Markieren Sie die Rollen der am Entwurfsmuster beteiligten Klassen und Methoden wie in (a) im Quelltext.
- Implementieren Sie zwei konkrete Heinzelmännchen:



- Kugel-Anmal-Heinzelmännchen*: Setzt die Farbe von Kugeln auf „rot“.
- Kerzen-Anzünde-Heinzelmännchen*: Setzt den Status jeder Kerze auf „brennend“.

Ergänzen Sie evtl. zusätzlich benötigte Eigenschaften in den konkreten Baumelementen.

² Der Einfachheit halber sehen wir Nadeln als integralen Bestandteil der Zweige an und modellieren sie hier nicht. Des Weiteren handelt es sich beim Stamm des Baumes auch um einen (überdimensionierten) Zweig.