



## Übungsblatt 07

### Aufgabe 17 (Zeiger- und Variablensemantik)

Eintrags-/Kollektions-Situationen lassen sich mit Zeiger- oder Variablensemantik modellieren. Betrachten Sie dazu die Fig. 5.26 und 5.27.

Definieren Sie jeweils in Zeiger- und Variablensemantik Module für Verwender, Eintrag und Kollektion in der textuellen Architekturnotation aus der Vorlesung.

Die Einträge sollen hier aus nur zwei Komponenten bestehen: `Schluesssel` vom Typ `SCHLUESSEL_T` und `Wert` vom Typ `WERT_T`. Modellieren Sie hierfür entsprechende Zugriffsoperationen.

Die Kollektion soll folgende Zugriffsoperationen anbieten:

1. Bewegen auf das erste Element
2. Bewegen auf das jeweils nächste Element
3. Abfrage des aktuellen Elements
4. Einfügen eines neuen Elements am Ende der Kollektion
5. Löschen des aktuellen Elements
6. Abfrage, ob ein aktuelles Element existiert

Der Verwender soll lediglich eine Operation `do` enthalten.

Implementieren Sie die Rümpfe der drei Module in beiden Varianten in einer Programmiersprache Ihrer Wahl (im Zweifel Ada). Die Kollektion soll als einfach verkettete Liste implementiert werden. Die Operation `do` des Verwenders soll folgendes tun:

1. Ein Element mit den Attributen `(x, 7)` anlegen.
2. Dieses Element zweimal in die Kollektion einfügen.
3. Ein Element mit den Attributen `(y, 8)` anlegen.
4. . 4. Dieses Element in die Kollektion einfügen.
5. Ein Element mit den Attributen `(x, 7)` anlegen.
6. Dieses Element in die Kollektion einfügen.
7. Das erste Element der Liste abfragen und seinen Wert auf 9 ändern.
8. Das erste Element der Liste entfernen.

Überlegen Sie, welche Unterschiede zwischen den beiden Varianten Zeiger- bzw. Variablensemantik auf Architekturebene sichtbar sind, d.h. welche Eigenschaften nicht nur innerhalb eines Modulrumpfs von Bedeutung sind, sondern auch von einem anderen verwendenden Modul beachtet werden müssen. An welchen Stellen reicht die textuelle Architekturnotation hier nicht aus, um diese Unterschiede zu beschreiben?

## Aufgabe 18 (Semantik von aktiv und passiv)

Rekapitulieren Sie die drei folgenden verschiedenen Semantiken von aktiv und passiv, die in der Vorlesung in Bezug auf Module bisher verwendet wurden, erläutern Sie sie und geben Sie jeweils ein Beispiel dafür an.

1. funktionale Bausteine vs. Datenabstraktionsbausteine
2. aktive Prozesse (Tasks) vs. passive Prozesse
3. Koordinierende und kontrollierte Bausteine

## Aufgabe 19 (Generizität und Vererbung)

Generizität als Mittel zur Wiederverwendung von Modulen wird in der Vorlesung im Rahmen des dort entwickelten Architekturmodells vorgestellt. Auf der anderen Seite werden im Zusammenhang mit der Objektorientierung Abstraktion und Spezialisierung von Klassen besprochen. Es ist möglich, diese Konzepte zu kombinieren, d.h. in einer objektorientierten Programmiersprache auch Generizität zu ermöglichen (z.B. C++, Java 1.5).

Eine weitergehende Kombination beider Konzepte stellt die *beschränkte Generizität* (constrained genericity) dar. Beispielsweise sei eine Klasse wie folgt angegeben (keine echte Syntax einer Programmiersprache!):

```
class List [ T -> Number ] {  
    ...  
}
```

Dies soll bedeuten, dass die Klasse `List`, die eine einfach verkettete Liste implementiert, mit einer Klasse `T` parametrisiert wird, die von `Number` abgeleitet sein muss. Die Elemente einer Instanz von `List[T]` müssen dann von der Klasse `T` oder einer Unterklasse von `T` sein. Bei der Implementierung von `List` können auf den Elementen alle Methoden von `Number` verwendet werden.

Überlegen Sie, welche Vorteile die Verwendung von beschränkter Generizität gegenüber unbeschränkter Generizität einerseits und dem völligen Verzicht auf Generizität andererseits mit sich bringt. Überlegen Sie dies zunächst am Beispiel einer Liste (wie oben). Können Sie sich in anderen Anwendungsbereichen weitere Vorteile vorstellen?

---

**Abgabe: Mittwoch, 21.06.2006**

Sie können Ihre Lösung zum obigen Termin in Papierform in der Vorlesung, der Übung oder in elektronischer Form per eMail an [pig@i3.informatik.rwth-aachen.de](mailto:pig@i3.informatik.rwth-aachen.de) abgeben.

Bitte vermerken Sie in jedem Fall die Namen und Matrikelnummern aller beteiligten Personen (maximal 3), beim Versenden per eMail auch alle eMail-Adressen. Abgaben in elektronischer Form können ausschließlich in den Dateiformaten *Plain-Text* oder *PDF* erfolgen.

Aktuelle Informationen zur Vorlesung finden Sie auf den Webseiten des Lehrstuhls unter <http://www-i3.informatik.rwth-aachen.de>.