
Aufgaben zur Klausur **Softwaredesign** und **Software Entwicklungs–Methoden** im SS 98
(WI h252, WI 56)

Zeit: 90 Minuten

erlaubte Hilfsmittel: keine

Bitte tragen Sie Ihre Antworten und fertigen Lösungen ausschließlich an den gekennzeichneten Stellen in das Aufgabenblatt ein. Ist ihre Lösung wesentlich umfangreicher, so überprüfen Sie bitte nochmals Ihren Lösungsweg.

Sollten Unklarheiten oder Mehrdeutigkeiten bei der Aufgabenstellung auftreten, so notieren Sie bitte, wie Sie die Aufgabe interpretiert haben.

Zwischen der Bearbeitung der Aufgaben 1 bis 8 über Softwaredesign mit Entwurfsmustern und den Aufgaben 9 und 10 über die Vorlesung SEM kann gewählt werden. Wird eine der Aufgaben 1 bis 8 bearbeitet, so werden die Aufgaben 1 bis 8 gewertet und Aufgaben 9 und 10 bleiben bei der Bewertung unberücksichtigt, sonst werden nur Aufgabe 9 und 10 gewertet.

Viel Erfolg !

Diese Klausur besteht einschließlich dieses Deckblattes aus 16 Seiten

Aufgabe 1:

Welches sind die Unterschiede zwischen klassenbasierten und objektbasierten Entwurfsmustern?

.....

.....

.....

Aufgabe 2:

Worin besteht der softwaretechnische Nutzen bei der Verwendung von Erzeugungsmustern?

Stichworte:

- 1)
- 2)
- 3)

Aufgabe 3:

Das Fliegengewicht ist ein Strukturmuster. Welcher Zweck wird mit dem Fliegengewichts-Muster verfolgt?

.....
.....
.....

Wann ist das Muster anwendbar?

- 1)
- 2)
- 3)

Warum erzeugt man Fliegengewichte nicht durch einen direkten Aufruf eines Konstruktors?

.....
.....
.....

Aufgabe 4:

Geben Sie zu der folgenden abstrakten Syntax ein gleichwertiges OMT-Diagramm an.

$$.0 \quad \textit{Expr} \quad = \textit{Ident} \mid \textit{Const} \mid \textit{UnExpr} \mid \textit{BinExpr}$$

$$.1 \quad \textit{Ident} \quad = \textit{String}$$

$$.2 \quad \textit{Const} \quad = \mathbb{Z}$$

$$.3 \quad \textit{UnExpr} \quad = \textit{Op} \times \textit{Expr}$$

$$.4 \quad \textit{BinExpr} \quad = \textit{Expr} \times \textit{Op} \times \textit{Expr}$$

Op und *String* sollen dabei für einfache vordefinierte Datentypen stehen.

Welche Klassen in diesem Modell könnten als Fliegengewicht verwaltet werden?

.....
.....
.....

Welche Strukturmuster kommen in dem Modell vor?

- 1)
- 2)
- 3)



Aufgabe 5:

Für die Verarbeitung von Baumstrukturen wird typischerweise das Kompositum-Muster eingesetzt. Geben Sie das OMT-Diagramm für die grundlegende Struktur des Kompositum-Musters an.

Mit welchen Mustern verarbeitet man Kompositum-Objekte typischerweise

- 1)
- 2)
- 3)

Aufgabe 6:

Im funktionalen Programmierstil, auch in Pascal und C, sind als Parameter auch Funktionen möglich. Diese werden bei ereignisgesteuerten Systemen häufig als call-back-Routinen verwendet. In objektorientierten Sprachen, z.B. Eiffel und Java, fehlt das Konzept der Funktionen als Parameter.

Mit welchem Verhaltensmuster werden Funktionen als Parameter simuliert?

.....

Geben Sie das OMT-Diagramm für die Struktur dieses Musters an.

Aufgabe 7:

Rekursive Datenstrukturen verarbeitet man typischerweise nach dem Interpretierer-Muster. Will man die Datenstruktur aber auf viele unterschiedliche Arten verarbeiten, wird das Interpretierer-Muster unflexibel. Warum?

.....
.....
.....

Welches Muster ist in diesem Fall besser geeignet?

.....

Begründung:

.....
.....



Aufgabe 9:

Entwickeln Sie mit den Methoden aus der Vorlesung ein hierarchisches Datenmodell für die Verwaltung von Musik-CDs.

Daten über CDs sollen in einem *CD-Katalog* gespeichert werden. Eine CD soll eindeutig durch ihren *Titel* und den *Interpreten* identifizierbar sein. Zu jeder CD sollen die folgenden Informationen verfügbar sein: ein *Untertitel*, wenn es einen gibt, das Plattenlabel, das Erscheinungsjahr, die *GesamtSpieldauer* in Sekunden und die Information über alle *Musikstücke*.

Zu jedem Musikstück sollen die folgenden Informationen vorhanden sein: der *Titel* des Stücks, der *Interpret*, falls er sich von dem der gesamten CD zugeordneten Interpreten unterscheidet, und die *Spieldauer* des Stücks in Sekunden.

Die Bereiche *Titel*, *Label* und *Interpret* seien als *Text* vorgegeben, mit *Text = Char**.

Für die Bezeichnung der einzelnen Datentypen verwenden sie bitte (auch) die kursiv geschriebenen Namen.

Das Datenmodell:

- 1)
- 2)
- 3)
- 4)
- 5)
- 6)
- 7)
- 8)
- 9)
- 10)
- 11)
- 12)

Geben Sie, wenn notwendig, die Konsistenzbedingungen an, die an dieses Datenmodell zu stellen sind, damit unlogische Wertekombinationen ausgeschlossen werden.

Die Konsistenzbedingungen in umgangssprachlicher Form:

.....

.....

.....

.....

.....

Die Konsistenzbedingungen formal:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Nehmen Sie dieses hierarchische Modell her und transformieren es in ein gleichwertiges, das 1-1 in ein Datenbankschema für eine relationale Datenbank umgesetzt werden kann, d.h. in den Abbildungen (maps) dürfen nicht wieder Abbildungen, Mengen oder Listen als Komponenten auftauchen, optionale Werte, kartesische Produkte und *Texte* bereiten keine Schwierigkeiten. Die Definition einfacher Bereiche braucht nicht wiederholt werden.

1)

2)

3)

4)

5)

6)

7)

8)

9)

10)

Begründen Sie ihre Lösung:

.....
.....
.....
.....
.....

Das oben entwickelte Datenmodell soll verfeinert werden. Die Informationen über die Interpreten auf den CDs soll feiner strukturiert sein: Der häufigste Fall ist zwar der, das genau ein Interpret einer CD zuzuordnen ist, es gibt aber auch Fälle, bei denen es mehrere Interpreten gibt (die 3 größten Tenöre), oder die Interpretenangabe fehlt (Bravo Hits 257). Ähnlich ist es bei einzelnen Stücken: die Interpreten einzelner Stücke können sich unterscheiden von denen der ganzen CD (Tina Turner Live: Duett mit Eric Clapton). Es kann aber durchaus die Reihenfolge der Interpreten von Bedeutung sein.

Modifizieren Sie die Datentypen aus dem ersten hierarchischen Modell an den erforderlichen Stellen.

Die zusätzlichen und die geänderten Datentypen:

- 1)
- 2)
- 3)
- 4)
- 5)
- 6)
- 7)

Begründen Sie ihre Lösung:

.....

.....

.....

.....

.....

Transformieren Sie dieses zweite, verfeinerte Modell auch wieder so, daß es 1-1 in ein Datenbankschema für eine relationale Datenbank umgesetzt werden kann, es gelten die gleichen Regeln wie oben:

- 1)
- 2)
- 3)
- 4)
- 5)
- 6)
- 7)
- 8)
- 9)
- 10)

Begründen Sie Ihre Lösung:

.....

.....

.....

.....

.....

Aufgabe 10:

Eine recht universell einsetzbare Datenstruktur zur Modellierung von hierarchischen und vernetzten Strukturen ist ein Objektsystem, in dem die Begriffe *Objekt* — *Attribut* — *Wert* eine zentrale Rolle spielen.

Ein Objektsystem soll in diesem Beispiel aus folgenden Komponenten aufgebaut sein: es gibt eine Menge von Objekten, jedes Objekt wird durch einen eindeutigen Namen identifiziert, jedem Objekt ist eine Menge von Attributen zugeordnet, jedes Attribut besteht aus einem Namen und einem Wert. Als Werte sind elementare Werte möglich und Verweise auf andere Objekte. Als elementare Werte sollen hier nur Texte zugelassen sein:

- .0 *ElementarerWert* = *Text*
- .1 *Text* = *Char**

Definieren Sie einen geeigneten Datentyp *ObjektSystem* für die oben beschriebene Struktur:

- 1)
- 2)
- 3)
- 4)
- 5)
- 6)
- 7)

Gibt es Konsistenzbedingungen, die beachtet werden müssen? Wenn ja, welche:
Umgangssprachlich:

.....

.....

.....

.....

.....

Formal:

.....

.....

.....

.....

.....

