

**Aufgabe 1: Assoziation, Aggregation, Komposition und Generalisierung**

Erläutern Sie die Eigenschaften, die besonderen Varianten und Unterschiede von Assoziation, Aggregation, Komposition und Generalisierung anhand folgender Beispiele. Wählen Sie für die einzelnen Teilaufgaben jeweils das aus Ihrer Sicht am besten geeignete Modellierungskonstrukt.

- a) Die Mitglieder einer Musikband setzen sich aus Sängern und Instrumentalisten zusammen. Instrumentalisten werden noch in Keyboarder, Gitarist und Bassist unterteilt.
- b) Eine Reise kann mehrere Ausflüge beinhalten, ein Ausflug kann für mehrere Reisen angeboten werden.
- c) Eine Ausstellung beinhaltet mindestens ein Gemälde, dabei kann es sich um ein Original oder um eine Kopie handeln. Ein Gemälde wird bei mehreren Ausstellungen gezeigt.
- d) Ein Forum besitzt einen Namen und eine Anzahl an Moderatoren. Ein Forum ist entweder ein Nachrichten- oder ein Diskussionsforum. In einem Forum existieren eine Vielzahl an Postings, ein Posting kann eine Antwort auf ein anderes Posting sein. Wird ein Posting gelöscht, werden auch sämtliche Antworten gelöscht.
- e) Ein Event kann von einer oder mehreren Firmen gesponsert werden. Eine Firma kann mehrere Events sponsern. Es ist bekannt, mit welchem Betrag eine bestimmte Firma ein bestimmtes Event sponsert.
- f) An einem Blind-Date nehmen immer zwei Personen teil. Eine Person kann zwar an mehreren Blind-Dates teilnehmen, doch die Kenntnis einer Person reicht nicht aus, um daraus auf alle seine/ihre Dates zu schließen.

## Aufgabe 2: Vergleich von Klassendiagrammausschnitten

Erklären Sie den Unterschied zwischen folgenden Klassendiagrammausschnitten.

a)



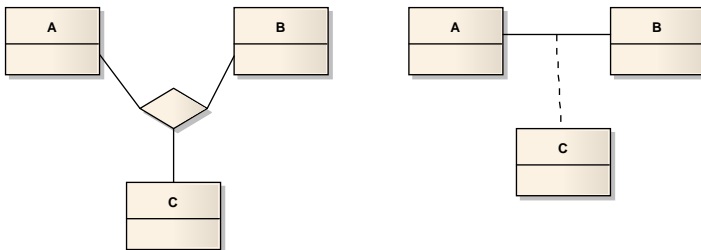
b)



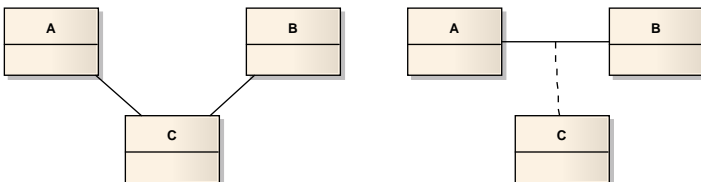
c)



d)



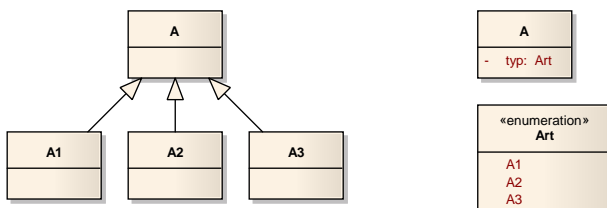
e)



f)

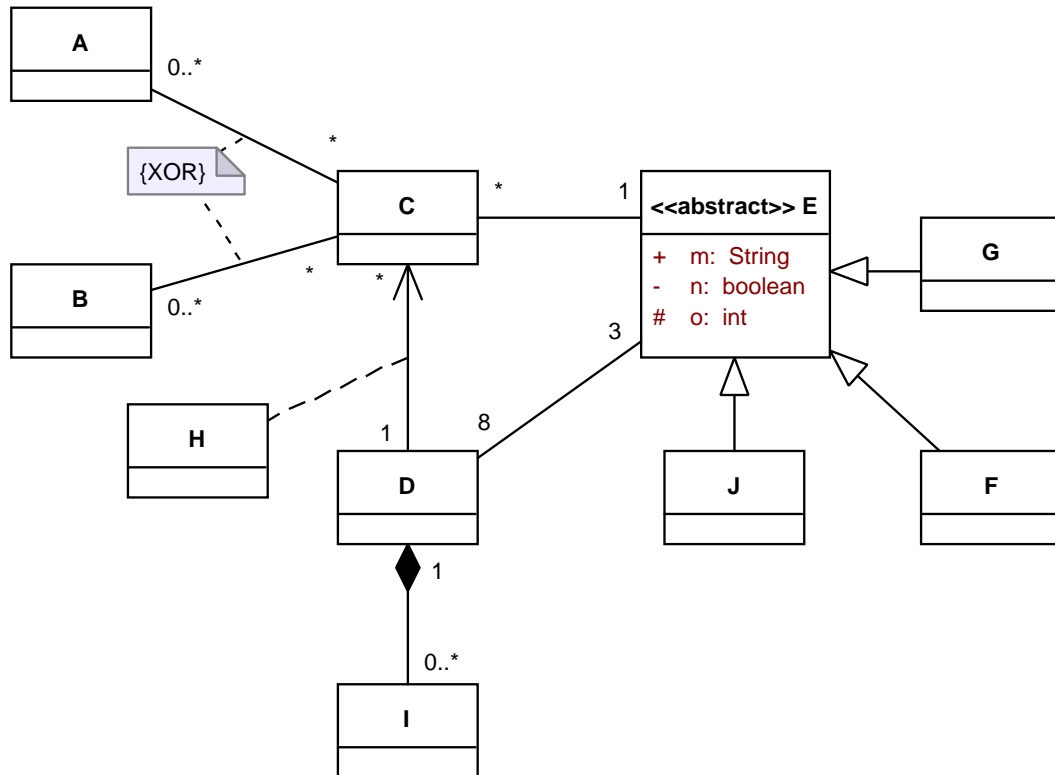


g)



### Aufgabe 3: Wahr oder falsch?

Es ist folgendes UML-Modell gegeben:



Welche Aussagen treffen zu? Begründen Sie Ihre Antwort!

Objekte von G und C können direkt auf o zugreifen.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von D ist in 0..* Objekten von I enthalten.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von C steht in Beziehung zu * Objekten von A oder * Objekten von B.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von E kann mit einem Objekt von C in Beziehung stehen.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt einer Subklasse von E kann mit 0 Objekten von C in Beziehung stehen.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch

Ein Objekt von F muss mit genau einem Objekt von E in Beziehung stehen.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von D steht in Beziehung zu genau einem Objekt von I.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Objekte von F und G können direkt auf o zugreifen.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Es gibt genau 8 Objekte von D, die in Beziehung zu genau 3 Objekten von E stehen.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch

Ein Objekt von C steht in Beziehung zu genau einem Objekt von E.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
J erbt alle Attribute von E.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von I kann in einem Objekt von D enthalten sein, muss aber nicht.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von G steht in Beziehung zu * Objekten von C.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch
Ein Objekt von D steht in Beziehung zu genau einem Objekt von C und dieses kann von D aus navigiert werden.	<input type="checkbox"/> richtig	<input type="checkbox"/> falsch

#### Aufgabe 4: Fehler finden

Gegeben ist folgendes UML Klassendiagramm. Bei der Modellierung sind leider einige Fehler passiert. Finden Sie 5 Fehler und korrigieren Sie diese im Diagramm.

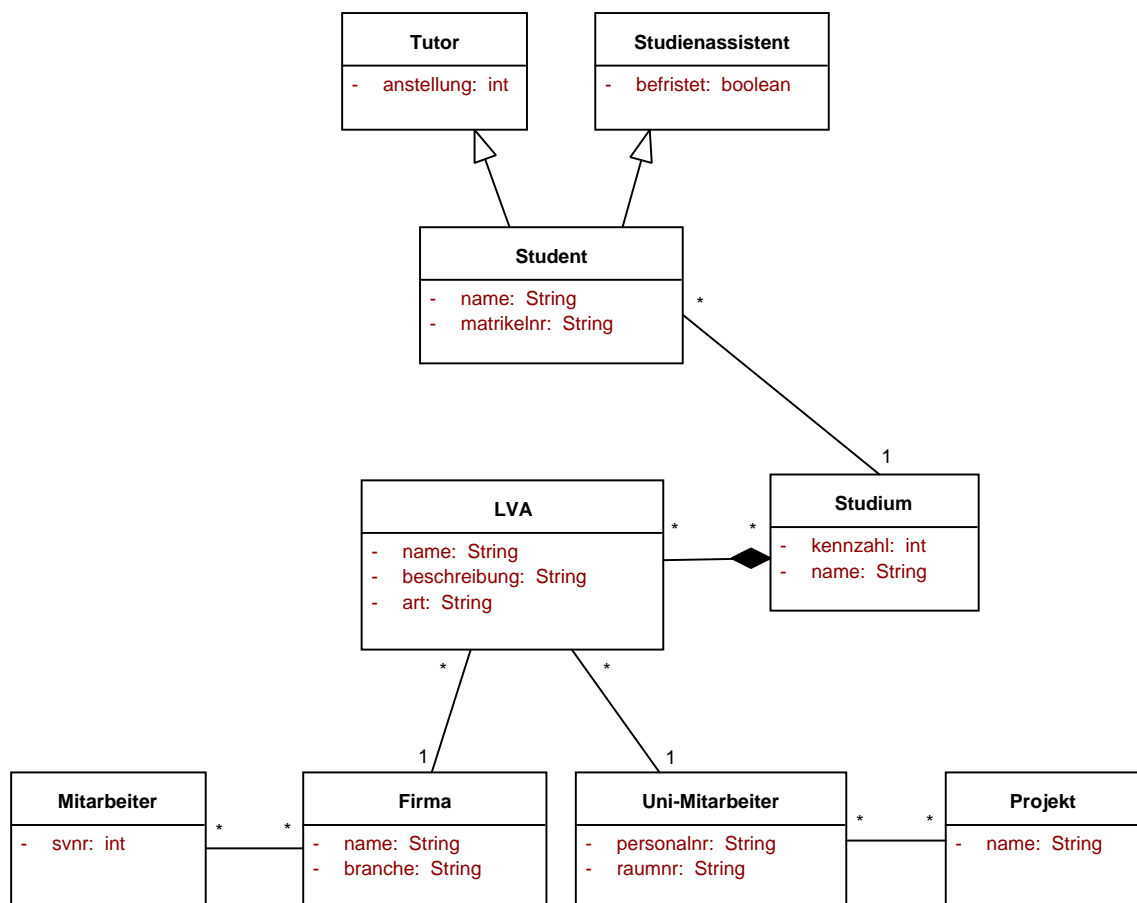
Jeder Student besitzt einen Namen sowie eine Matrikelnummer. Ein Student kann zusätzlich als Tutor oder Studienassistent an der Universität tätig sein. Bei Tutoren wird zusätzlich die Anstellung gespeichert, bei Studienassistenten ist wichtig, ob diese befristet sind oder nicht.

Ein Student kann verschiedene Studien inskribieren, ein Studium kann von vielen Studenten absolviert werden. Von jedem Studium sind die Kennzahl und der Name bekannt. Das Datum, zu dem sich ein bestimmter Student für ein bestimmtes Studium einschreibt, wird gespeichert.

Ein Studium setzt sich aus mehreren LVAs zusammen, jede LVA besitzt einen Namen, eine Beschreibung und eine bestimmte Art. Eine LVA ist Teil eines Studiums, sie kann in unterschiedlichen Studien vorkommen.

Die Inhalte einer LVA werden entweder von einer Firma, von der Name und Branche bekannt sind, oder von einem Universitäts-Mitarbeiter, der eine Personal- sowie Raumnummer besitzt, bereit gestellt. Universitäts-Mitarbeiter können bei mehreren Projekten mitarbeiten, jedes Projekt besitzt einen Namen.

Eine Firma beschäftigt mehrere Mitarbeiter, ein Mitarbeiter kann bei mehreren Firmen angestellt sein. Jeder Mitarbeiter hat eine eindeutige Sozialversicherungsnummer.



### **Aufgabe 5: Klassendiagramm modellieren**

Modellieren Sie folgenden Sachverhalt als UML-Klassendiagramm. Geben Sie in Ihrem Diagramm die beschriebenen (abstrakten) Klassen und ihre Attribute (inklusive Typen), sowie die Assoziationen (inklusive Multiplizitäten, Aggregationen, etc.) an.

Ein Laufclub besitzt eine Bezeichnung sowie ein Gründungsjahr. Jeder Laufclub hat viele Läufer als Mitglieder, von jedem Läufer sind der Name und das Geburtsdatum bekannt. Läufer können in genau zwei Gruppen unterteilt werden: Hobbyläufer und Profiläufer. Von einem Profiläufer sind die Anzahl der bestrittenen Marathonläufe sowie die beste Marathonzeit bekannt.

Ein Laufclub veranstaltet mehrere Läufe, jeder Lauf besitzt neben einem Namen und einem Datum auch eine Distanz, die zurückgelegt werden muss.

An einem Lauf nehmen mehrere Läufer teil, ein Läufer kann bei mehreren Läufen an den Start gehen. Es wird für jeden bestimmten Läufer, der einen bestimmten Lauf absolviert, die benötigte Zeit gespeichert.

Ein Lauf wird von mehreren Sponsoren finanziert. Es ist bekannt, wie viel ein bestimmter Sponsor für einen bestimmten Lauf an Budget zur Verfügung stellt. Ein Sponsor besitzt einen Namen und kann neben Läufen auch mehrere Profiläufer sponsern, jeder Profiläufer kann sich von mehreren Sponsoren unterstützen lassen.

Ein Lauf kann entweder im Zuge eines Laufcups oder im Zuge eines Lauftreffs abgehalten werden. Ein Lauftreff wird von einem Laufclub organisiert, ein Laufcup hat eine bestimmte Bezeichnung.

### **Aufgabe 6: Sight-Seeing**

Modellieren Sie folgenden Sachverhalt mittels UML Klassendiagramm gemäß folgenden Informationen:

Jede Stadt besitzt mehrere Sehenswürdigkeiten. Eine Stadt hat einen Namen und eine bestimmte Einwohnerzahl, eine Sehenswürdigkeit hat eine Bezeichnung sowie eine Adresse. Für eine Sehenswürdigkeit werden verschiedene Führungen angeboten, von einer Führung sind die Dauer und die Nummer bekannt.

An einer Führung können maximal 20 Besucher teilnehmen, bestimmten Besuchern kann bei bestimmten Führungen eine Ermäßigung gewährt werden, die Höhe dieser Ermäßigung muss bekannt sein. Eine Führung benötigt natürlich auch einen Leiter. Bei den Besuchern und dem Leiter der Führung handelt es sich um Personen, die einen Namen besitzen.

### **Aufgabe 7: Modellierung Kreuzfahrt I**

Eine Kreuzfahrt wird von einem bestimmten Anbieter veranstaltet, jeder Anbieter besitzt einen Namen und einen Standort. Eine Kreuzfahrt ist durch eine Nummer sowie eine Bezeichnung gekennzeichnet. Für eine Kreuzfahrt benötigt man viele Mitarbeiter, jeder dieser Mitarbeiter besitzt einen Namen und eine Sozialversicherungsnummer.

Bei Mitarbeitern unterscheidet man zwischen dem Kapitän, den Offizieren und dem sonstigen Personal. Jeder Kapitän hat eine gewissen Anzahl an Dienstjahren geleistet, ein Offizier besitzt einen Rang – erster, zweiter oder dritter Offizier – eine andere Einstufung ist nicht möglich. Bei allen sonstigen Mitarbeitern wird die Abteilung gespeichert, in der sie arbeiten.

An einer Kreuzfahrt nehmen mehrere Gäste teil, jeder Gast hat einen Namen und es wird zusätzlich vermerkt, ob es sich um einen VIP handelt oder nicht. Ein Gast kann natürlich auch mehrere Kreuzfahrten unternehmen. Von jedem Schiff ist der Schiffsname, die Anzahl der Decks, die Länge des Schiffs sowie die Anzahl der Passagiere bekannt. Ein Schiff kann für verschiedene Kreuzfahrten genutzt werden.

Modellieren Sie diesen Sachverhalt mittels UML Klassendiagramm.

## Aufgabe 8: Modellierung Kreuzfahrt II

Das vorhergehende Beispiel wird wie folgt ergänzt:

Eine Kreuzfahrt kann entweder in einem Reisebüro oder über eine Website gebucht werden. Jede Buchung hat eine eindeutige Buchungsnummer und ein Buchungsdatum. Ein Reisebüro besitzt einen Namen und eine Adresse, eine Website hat eine Domain, über die sie erreichbar ist.

Eine Kreuzfahrt steuert immer ein bestimmtes Zielgebiet an, jedes Zielgebiet hat eine Bezeichnung, darüber hinaus wird der Name des Meeres vermerkt. Auf einer Kreuzfahrt werden den Gästen mehrere Shows geboten, jede Show hat einen Titel und eine gewisse Dauer.

Ergänzen Sie Ihr UML Klassendiagramm entsprechend.

## Aufgabe 9: Reverse Engineering

Gegeben sei der unten angeführte Java ähnliche Code. Führen Sie ein Reverse Engineering des Codes in ein UML Klassendiagramm durch. Das heißt, Sie müssen ein UML Klassendiagramm entwerfen, das semantisch dem Java Code entspricht. Bilden Sie Referenzen möglichst durch Assoziationen ab.

```
class Telefonbuch {
    public String name;
    private Typ typ;
    private Person besitzer;
    public Eintrag [] eintraege;
    public Hashtable berechtigungen;
        // Key: Person (Typ: Person)
        // Value: Berechtigung (Typ: String)

    void addEintrag (Eintrag eintrag) {...}
    Eintrag getEintrag (String name) {...}
    void setBerechtigung (Person p, String art) {...}
}

Enumeration Typ {
    privat;
    geschaeftlich;
}

class Person {
    public String name;
    public Telefonbuch [] telefonbuecher;
}

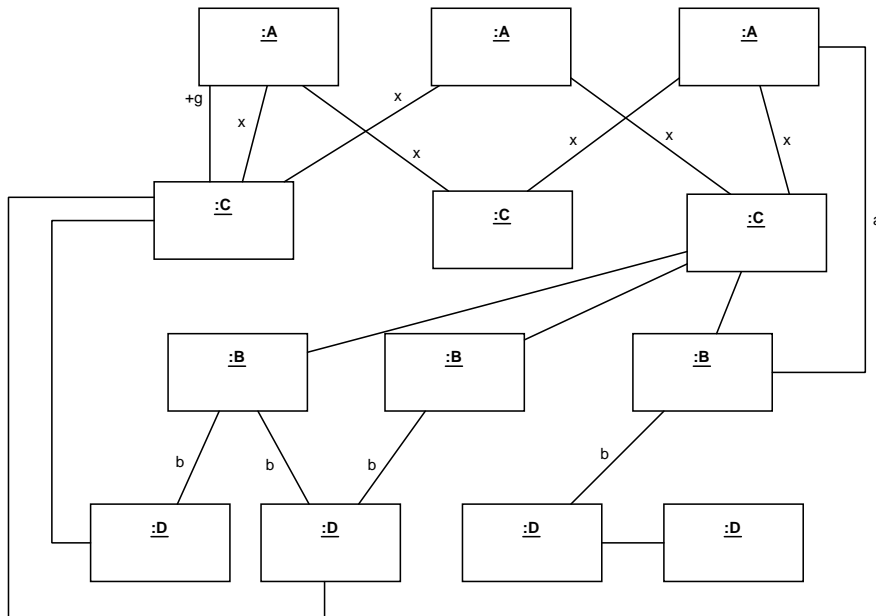
class Freund extends Person {
    public Date geburtsdat;
}

class Eintrag {
    Person person;
    public int telnr;
    public String adr;
}
```

### Aufgabe 10: Objektdiagramm

Entwerfen Sie zwei Klassendiagramme, zu denen nachfolgende Objektdiagramme konform sind. Wählen Sie die Kardinalitäten an den Assoziationsenden möglichst genau. Sie können davon ausgehen, dass diese Objektdiagramme die höchstzulässige Anzahl an Beziehungen mit Objekten einer anderen Klasse darstellen. Weiters sollen mögliche Generalisierungen bzw. XOR-Beziehungen erkannt werden.

- Objektdiagramm 1:



- Objektdiagramm 2:

