

Übungsblatt 2

Modellierung

Im Fach Informatik ist die Modellierung ein wichtiger Bestandteil des Entwicklungsprozesses von Soft- und Hardware. Ein Modell dient dazu, eine Problemstellung genau zu untersuchen und als Ganzes oder in Teilen zu beschreiben. Es ist die Vorstufe für den Entwurf von Algorithmen, Software, Datenbanken und Hardware.¹

Aufgabe 1 - Modelle aus dem Alltag

Geben Sie drei Modelle aus Ihrem Alltag an und schreiben Sie zu jedem Modell auf

- a) Was wird modelliert?
- b) Welchem Zweck dient das Modell?
- c) Was sind die wichtigen Objekte, Eigenschaften, Zusammenhänge und Operationen des Modells?

UML

Für die Modellierung insbesondere von objektorientierter Software nutzt man in der Informatik die Modellierungssprache UML.

Aufgabe 2 - UML-Diagrammtypen

Beschreiben Sie den Unterschied zwischen Struktur- und Verhaltensdiagrammen. Geben Sie jeweils ein Beispiel in UML.

Aufgabe 3 - Objektbeziehungen

Beschreiben Sie jeweils mit Hilfe eines UML-Klassendiagramms, die Beziehungen und deren Multiplizitäten zwischen den Objekten in den Folgenden Beispielen. Um welche Art von Beziehung handelt es sich jeweils? Überlegen Sie, wie Sie diese Beziehungen jeweils in der Objektorientierten Programmiersprache C++ umsetzen könnten.

1. 'besitzt' zwischen Kunde*in (einer Bank) und Konto (dieser Bank).
2. 'verheiratet' zwischen zwei Personen.
3. zwischen Hörsaal und Gebäude.
4. zwischen Vorlesung und Student*in.

¹Kastens, U., & Büning, H. K. (2018). Modellierung: Grundlagen und formale Methoden. Carl Hanser Verlag GmbH & Co KG.

Aufgabe 4 - Gerichtete und ungerichtete Assoziation

In der Vorlesung wird eine gerichtete Assoziation an folgendem C++ Beispiel gezeigt:

```
#include <set>

class Firma {

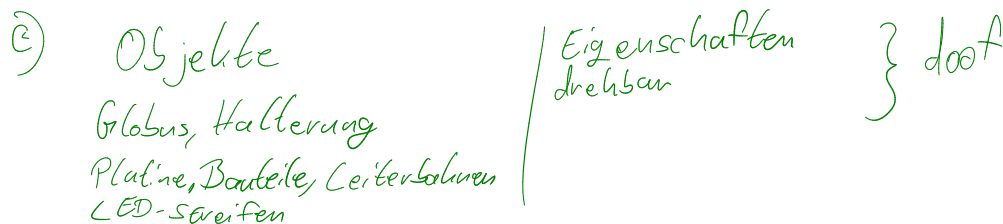
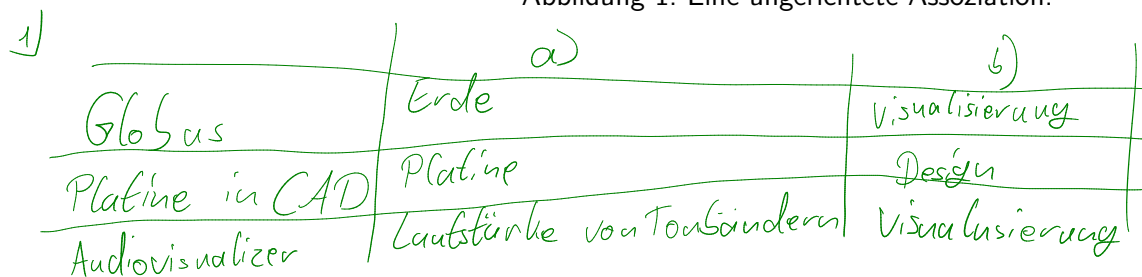
public:
    void schickt_Werbung()
    {
        for(Person *kunde : kunden)
        {
            /* schicke werbung an kunde */
        }
    }

private:
    std::set<Person*> kunden;
};
```

Setzen Sie analog die ungerichtete Assoziation aus Abbildung 1 in C++ Quelltext um.



Abbildung 1: Eine ungerichtete Assoziation.



2)

Struktundiagramme sind statisch während Verhaltensdiagramme dynamisch sind

Struktundiagramm: Klassendiagramm, Objektdiagramm

Verhaltensdiagramm: Anwendungsfall-, Aktivitäts-, Zustandsdiagramm

