

Bearbeiten Sie alle Aufgaben so, dass Sie für die Übungen **gut vorbereitet** sind, d.h. dass Sie Ihre Lösungen in der Übungsgruppe in geeigneter Form **zeigen, erklären** und **diskutieren** können. Geben Sie stets Ihre verwendeten **Quellen** an.

Aufgabe 3-1: UML-Elemente

Lernziel: Wichtige UML-Modellelemente kennen, lesen und verwenden können.

Betrachten Sie die folgenden UML-Modellelemente:

Klassen und Objekte, Attribute und Operationen, Assoziationen, Multiplizität, Generalisierung, sowie Komposition und Aggregation.

Beantworten Sie für *jedes* Element die folgenden Fragen:

- a) Was ist der Zweck bzw. die Semantik des Elements? Was sind die Beziehungen zu den anderen genannten Elementen?
- b) Wie wird das Element grafisch oder textuell notiert? Welches sind dabei die notwendigen Notationsbestandteile und welches die optionalen? Beachten Sie: Die meisten Modellelemente haben sehr viele optionale Notationsbestandteile!

Sie können für die Bearbeitung dieser Aufgabe die UML-Spezifikation selbst heranziehen (siehe Aufgabe 3-3). Sie dürfen aber auch andere, leichter zu lesende Quellen verwenden. Denken Sie wie immer daran, diese auch anzugeben.

Aufgabe 3-2: UML-Klassendiagramme

Lernziel: In natürlicher Sprache ausgedrückte, statische Sachverhalte vollständig und präzise als Klassenmodell erfassen und als UML-Klassendiagramm darstellen können.

Modellieren Sie den folgenden Sachverhalt möglichst vollständig und stellen Sie das Ergebnis als UML-Klassendiagramm dar. *Vollständig* heißt dabei, dass alle Aussagen aus dem Beschreibungstext modelliert wurden. Achten Sie auch darauf, dass das Klassendiagramm nicht *mehr* aussagt, als der Text hergibt.

Im gefährlichen Trollwald gibt es viele Höhlen. Jede Höhle hat eine bestimmte Anzahl von Eingängen und eine bestimmte Tiefe. Jede dieser Höhlen dient als Behausung für einen oder mehrere Trolle. Es gibt im Trollwald zwei verschiedene Arten von Trolle: Steintrolle und Bergtrolle. Alle Trolle wohnen in Höhlen und sind durch ihre Nasenlänge und ihre Größe unterscheidbar. Steintrolle bestehen aus einer bestimmten Gesteinsart. Bergtrolle bestehen zwar nicht aus Stein, jedoch haben sie Schuppen in den schillerndsten Farben. Jeder Troll hat natürlich eine Keule, mit der er hauen kann und die sein Eigentum ist. Tolle Trolle haben sogar mehrere solcher Keulen. Die Keulen sind entweder aus Eiche oder aus Fichtenholz.

Aufgabe 3-3: UML-Metamodell

Lernziel: Sich in strukturierten technischen Dokumenten zurechtfinden können (nicht entmutigen lassen; die Aufgabe ist an sich nicht schwer); Unterschied zwischen Diagramm und Modell, sowie mehrere Abstraktionsebenen verstehen.

Hintergrund: Konkrete UML-Diagramme (z.B. Ihr Ergebnis aus Aufgabe **3-2**) bestehen aus grafischen Elementen, die jeweils Elemente des zu Grunde liegenden UML-Models repräsentieren. Die prinzipiell verfügbaren Modellelemente (siehe z.B. Aufgabe **3-1**) werden im sog. UML-Metamodell als Metaklassen (= Klassen im Metamodell) spezifiziert.

Jedes der Kapitel 7 bis 20 der UML-Spezifikation beleuchtet einen UML-Themenbereich. Konkrete Metaklassen beschreiben je eine Sorte von UML-Modellelementen, die in tatsächlichen UML-Modellen und -Diagrammen vorkommen (wie etwa *Klassen*, *Assoziationen* oder *Generalisierungen*). Abstrakte Metaklassen sind hingegen nur technische Hilfsmittel im UML-Metamodell und werden in konkreten UML-Modellen und -Diagrammen nicht direkt verwendet.

Die automatisch generierten Abschnitte "Classifier Descriptions" jedes der Kapitel 7 bis 20 listen alle Metaklassen auf: Konkrete Metaklassen haben den Suffix "[Class]", abstrakte Metaklassen sind mit "[Abstract Class]" markiert.

Aufgabe: Laden Sie zur Bearbeitung der Teilaufgaben die Spezifikation des UML-Metamodells in der aktuellen Version 2.5.1 herunter: <http://www.omg.org/spec/UML/2.5.1/PDF/>.

- a) Die folgenden UML-Modellelemente werden v.a. in *Klassendiagrammen* verwendet. Finden Sie die zugehörigen (konkreten) Metaklassen in der Spezifikation und geben Sie jeweils den Namen der Metaklasse und die relevanten Abschnittsnummern an.

Hinweis: Kapitel 9-11 sollten hierbei besonders hilfreich sein.

Beispiel: Normale Klassen aus einem UML-Modell werden in Diagrammen grafisch als Kästchen dargestellt (wie in Aufgabe **3-2**). Technisch gesehen sind sie Exemplare vom Typ `Class` aus dem UML-Metamodell, der in Abschnitt 11.4 beschrieben und in 11.8.3 zusammengefasst wird.

UML-Modellelement	Name der Metaklasse	Relevante Abschnitte
Klasse Objekt (Exemplar einer Klasse) Generalisierung Attribut Operation Komposition & Aggregation Schnittstelle Implementierung (einer Schnittstelle) Assoziation	<code>Class</code>	11.4 und 11.8.3

- b) Die folgenden UML-Modellelemente finden sich in anderen Diagrammtypen wieder. Finden Sie auch hier die zugehörigen Metaklassen heraus.

Hinweis: Kapitel 13 und 14 sollten hierbei besonders hilfreich sein.

Modellelement	Diagrammtyp	Metaklasse	Relevante Abschnitte
Methodenaufruf	Sequenzdiagramm		
Interner Übergang	Zustandsdiagramm		
Externer Übergang	Zustandsdiagramm		

- c) Die Knoten in Aktivitätsdiagrammen können entweder *atomare Aktionen* oder *zusammengesetzte Aktivitäten* sein. In UML sind eine Reihe von *Aktionen* vordefiniert.

Hinweis: Kapitel 16 sollte hierbei besonders hilfreich sein.

Finden Sie zu drei dieser Aktionen jeweils die Metaklasse heraus und beschreiben in eigenen Worten, was diese Aktion jeweils tut.

Aufgabe 3-4★: UML-Sequenzdiagramme

- a) Was ist ein Sequenzdiagramm und wozu dient es im Kontext der Softwareentwicklung?
- b) Welche Modellelemente werden sowohl in Klassendiagrammen als auch in Sequenzdiagrammen benutzt?
- c) Was ist in der Vertikalen dargestellt?
- d) Wie wird ein Aufruf einer Methode eines Objektes dargestellt?
- e) Ist eine Nachricht das gleiche wie ein Methodenaufruf?
- f) Wie kommen die Objekte "aneinander heran", also worin besteht die Verbindung zwischen Sender-Objekt und Empfänger-Objekt, über die eine Nachricht gesendet wird?
- g) Woran ist klar zu erkennen, dass Sequenzdiagramme immer nur einen von vielen möglichen Abläufen darstellen?
- h) Geben Sie für die 15 mit Pfeilen gekennzeichneten Diagrammelemente jeweils die Bezeichnung an bzw. wofür diese stehen. Beispiel: 1 → *Diagrammname*

