

# Prüfung Software Engineering (SE)

Prof. Dr.-Ing. Christian Heller <christian.heller@ba-leipzig.de>

Student	
Vor- und Nachname	Ihre Daten werden von der Klausuraufsichtsperson schriftlich auf der Anwesenheitsliste festgehalten.
Matrikelnummer	
Studienrichtung und Jahr	CS 2018-1
Anmeldename	Das Login "klaus..." muss unbedingt schriftlich auf der Anwesenheitsliste festgehalten werden, da sonst keine Zuordnung des Logins zu Ihrem Namen und damit keine Korrektur der Klausur möglich ist!

Prüfung	
Datum	Juni 2020
Dauer [min]	100
Hilfsmittel	Dokumentation im lokalen Netzwerk (Intranet) sowie Recherche im Internet. NICHT gestattet: * Kommunikation in jeglicher Form * Anmeldung via SSH auf dem Rechner "fileserv" * Anmeldung mit Klausur-Login nach Ende der Prüfung Dies kann leicht geprüft werden (last cs16*, Server-Log-Dateien). Bitte unterlassen Sie also Täuschungsversuche in Ihrem eigenen Interesse.
Bemerkungen	Hinterlegen Sie alle Programme und Antworten in elektronischer Form! Es wird kein Papier angenommen. Möchten Sie Lösungen erläutern, so nutzen Sie Quelltext-Kommentare oder legen eine Text-Datei an. Speichern Sie sämtliche Daten im HOME-Verzeichnis des Nutzers, d. h. unter Windows auf Laufwerk H:\ (NICHT auf C:\ oder "Eigene Dateien")! Idealerweise legen Sie dort ein Unterverzeichnis namens "klausur" an. Lesen Sie die Aufgaben komplett durch, bevor Sie sie lösen! Die Reihenfolge der Lösung ist Ihnen überlassen. Probieren Sie immer, eine Aufgabe zu lösen, da auch auf richtige Teile nicht vollständiger Lösungen Punkte vergeben werden! Falls vom Prinzip her richtig, so werden auch alternative Lösungen akzeptiert. Sie dürfen beliebig viele Bildschirmausgaben von Werten in den Quelltext einbauen, um ein Programm besser nachvollziehen zu können. Bitte duplizieren Sie Ihre Quelltextdateien (workspace) NICHT, da beim Korrigieren dann beide durchsucht werden müssen, was sinnlosen Aufwand verursacht. Diese Aufgabenstellung in Papierform können Sie nach dem Ende der Klausur behalten.

Bewertung						
Aufgabe	1	2	3	4	5	Summe
Punkte	20	10	20	20	30	100

Im Rahmen der Klausur sind Aufgaben mit Bezug zur Unified Modeling Language (UML) zu lösen. Insofern modelliert werden soll, ist das BOUML-Werkzeug zu verwenden.

## Aufgabe 1: Klassendiagramm „Eisessen“ [20]

Zweck: Modellieren eines Sachverhaltes mit Strukturelementen und Beziehungen.

Hinweis: Soweit wie möglich sollen die in der Beschreibung benutzten Begriffe verwendet werden. Auch Kardinalitäten sind, wo zutreffend, anzugeben.

- Es gibt Eisbehältnisse. Ein Eisbecher ist ein spezielles solches, eine Eistüte ebenfalls. [2]
- Desweiteren gibt es Eis, wobei Fruchteis und Joghurteis jeweils spezielles Eis ist. [2]
- Im Zusammenhang mit Eisbehältnissen kann solches Eis die Rolle des Inhaltes spielen, wobei maximal zehn Eis(-kugeln) in einem Eisbehältnis enthalten sein können. [2]
- Ein Eisbecher kann viele Früchte enthalten, die in Bezug auf den Eisbecher Beilagen sind. [2]

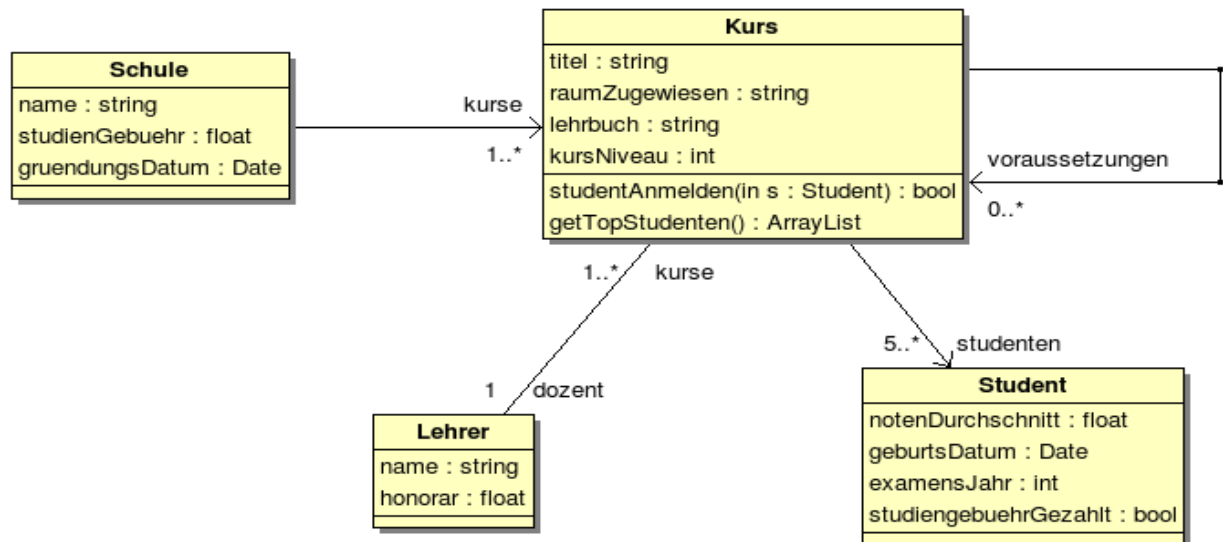


- e) Solche Früchte sind zum Beispiel Erdbeere und Sauerkirsche. [2]
- f) Weiterhin kann in einem Eisbecher eine Flüssigkeit enthalten sein, welche die Soße charakterisiert. [2]
- g) Solche Flüssigkeiten sind zum Beispiel Amaretto und Vanillesoße. [2]
- h) Verschiedene Preise eines Eisbehältnisses werden durch einen Geldwert charakterisiert: jedes Eisbehältnis (inklusive Inhalt) hat genau einen Verkaufspreis sowie einen Zutateneinkaufspreis. [2]
- i) Eine Person kann beliebig viele Eisbehältnisse besitzen, die für die Person Erfrischungsspeisen repräsentieren. [2]
- j) Eine Person kann die Tätigkeit des Genießens von Eis ausführen. [2]

## Aufgabe 2: Object Constraint Language (OCL) „Schule“ [10]

Zweck: Auswerten von OCL-Ausdrücken im Bezug auf eine Bildungseinrichtung.

Hinweis: Die jeweilige Interpretation des OCL-Ausdrucks zum gegebenen Klassendiagramm soll kurz in Worten erläutert werden. Schreiben Sie die Antwort in eine einfache Textdatei!



a) [2]

```
context Kurs inv: self.dozent.honorar > 0.00
```

b) [2]

```
-- Constraint given explicitly.
context s: Student
inv: s.notenDurchschnitt > 2.0
```

c) [2]

```
context Kurs::studentAnmelden(in s : Student) : bool
pre: s.studiengebuehrGezahlt = true
post: result = true and self.studenten = self.studenten@pre + 1
```

d) [2]

```
context Schule::studienGebuehr : float
init: 2500.00
```

e) [2]

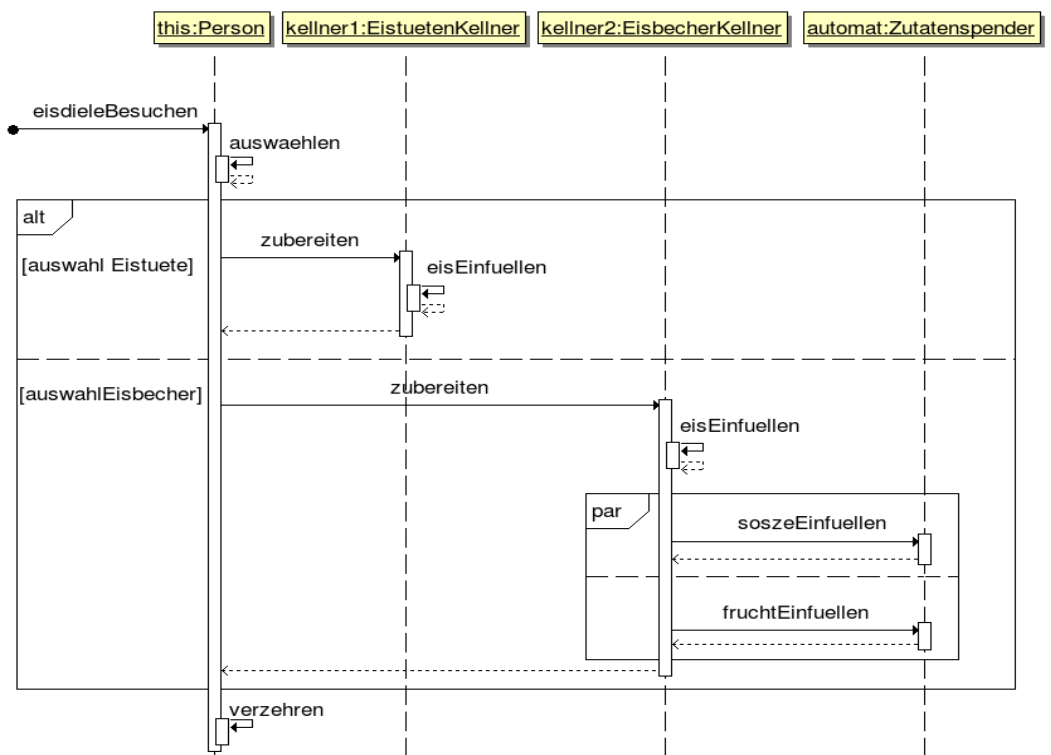
```

context Kurs inv:
let honorar : float = self.dozent.honorar in
if self.kursNiveau > 4000 then
  honorar > 80000.00
else
  honorar < 80000.00
endif
  
```

### Aufgabe 3: Sequenzdiagramm und Kommunikationsdiagramm „Eisdiele“ [20]

Zweck: Überführen Sequenzdiagramm in verhaltensgleiches Kommunikationsdiagramm.

Hinweis: Gegeben sei das Sequenzdiagramm. Im zu erstellenden Kommunikationsdiagramm soll hierarchische Nummerierung verwendet werden. Wenn das CASE-Werkzeug die Kennzeichnung alternativer oder parallel versendeter Nachrichten nicht unterstützt, kann ein Notizelement zur Beschreibung in Worten eingefügt werden.



- Objekt vom Typ „Person“ [2]
- Objekt vom Typ „Zutatenspender“ [2]
- Objekt vom Typ „EistuetenKellner“ [2]
- Objekt vom Typ „EisbecherKellner“ [2]
- Methoden an „Person“ [2]
- Methoden an „EistuetenKellner“ [2]
- Methoden an „EisbecherKellner“ [2]
- Methoden an „Zutatenspender“ [2]
- Hierarchische Nummerierung [2]
- Fragment [2]

#### **Aufgabe 4: Anwendungsfalldiagramm „Spielplatz“ [20]**

Zweck: Konzipieren eines Spielplatzes.

- a) Auf dem Spielplatz soll man schaukeln und rutschen können. [2]
- b) Es gibt zwei Varianten des Rutschens: Stange und Rutsche. [2]
- c) Als weitere Spielmöglichkeit gibt es das Klettern, welches bei Vorhandensein einer Aufstiegsrampe zum Hangeln erweitert werden kann. [2]
- d) Ein Kind als Akteur steht mit den drei vorgenannten Funktionalitäten in Beziehung. [2]
- e) Das Kind und auch der Platzwart sind Spezialisierungen von Mensch. [2]
- f) Des Platzwartes Aufgabe ist das Pflegen des Spielplatzes. [2]
- g) Das Pflegen umfasst Säubern und Reparieren. [2]
- h) Die Organisation TÜV ist für das Prüfen zuständig. [2]
- i) Alle Funktionalitäten gehören zum System Spielplatz. [2]
- j) Da es um Kinder geht, sollen Anwendungsfälle kunterbunt sein (mindestens drei Farben). [2]

#### **Aufgabe 5: Zustandsmaschinendiagramm „Volksabstimmung“ [30]**

Zweck: Darstellen der Etappen einer Volksabstimmung.

Erläuterung: Zu den direktdemokratischen Elementen zählt die Bundesweite Volksabstimmung. Die dreistufige Volksgesetzgebung gibt Bürgern das Recht, selbst Gesetzesinitiativen zu starten und bei genügend Unterschriften einen verbindlichen Volksentscheid dazu herbeizuführen:

<https://www.mehr-demokratie.de/themen/volksabstimmungen/bundesweite-volksabstimmung/>

Hinweis: Alle im folgenden Sachverhalt beschriebenen Schritte sind als Zustand (NICHT Aktion) zu modellieren.

- a) Bürgerinnen und Bürger erarbeiten einen Gesetzentwurf. [2]
- b) Anschließend starten sie eine Volksinitiative. [2]
- c) Dass es dafür keine Sammlungsfrist gibt, wird als Notizelement hinterlegt. [2]
- d) Kommt die nötige Anzahl an Unterschriften nicht zusammen, so endet die Volksinitiative. [2]
- e) Kommen die 100.000 Unterschriften jedoch zusammen, so wird der Vorschlag im Bundestag behandelt. [2]
- f) Als Randnotiz wird angemerkt, dass (1) der Vorschlag binnen 6 Monaten zu behandeln ist und (2) die Bürgerinitiative ein Rederecht erhält. [2]
- g) Hat der Bundestag Zweifel an der Verfassungsgemäßheit des Vorschlages, so ruft er das Verfassungsgericht an, was als vorläufiges Ende (exit) der Zustandsmaschine gilt. [2]
- h) Ist der Vorschlag legitim, so kann der Bundestag zustimmen, was in einer Gesetzgebung resultiert. [2]
- i) Die Gesetzgebung einschließlich aller Schritte des Verfahrens wird anschließend gespeichert (historisiert). [2]
- j) Wurde der Vorschlag durch den Bundestag abgelehnt, so kann die Bürgerinitiative ein Volksbegehren starten. Die Sammlungsfrist dafür beträgt 9 Monate, was zu notieren ist. [2]
- k) Kommen die geforderten 1 Mio. Unterschriften nicht zusammen, endet das Volksbegehren. Anderenfalls kommt es zum Volksentscheid, welcher als Unterzustandsmaschine zu modellieren ist. [2]

l) Bei Ablehnung endet der ganze Vorgang. Bei Zustimmung zum Volksentscheid ist der Bundestag gezwungen, den Vorschlag in ein Gesetz münden zu lassen. Dafür soll die Transition zurückverweisen auf den weiter oben bereits modellierten Zustand "Gesetzgebung". [2]

m) Die Unterzustandsmaschine "Volksentscheid" soll in einem eigenen Diagramm verfeinert werden. Ihr Eintrittspunkt soll heißen "Vorschlag der Bürger". [2]

n) Als zweiter möglicher Ausgangspunkt soll ein empfangenes Signal "Alternativvorschlag des Bundestages" dienen. Dieses ist mit der vom Eintrittspunkt kommenden Transition per Kreuzung (junction) zu verschmelzen. [2]

o) Schließlich folgen die Zustände "Versand Abstimmungsheft" und "Entscheid durch einfache Mehrheit", bevor die Unterzustandsmaschine am Exit Point "Abstimmungsergebnis" endet. [2]

**Viel Erfolg!**