



**BERGISCHE
UNIVERSITÄT
WUPPERTAL**

Prof. Dr. Hans-Jürgen Buhl
Praktische Informatik/Numerik

Fachbereich C
Mathematik und Naturwissenschaften,
Mathematik und Informatik

E-MAIL buhl@math.uni-wuppertal.de

WWW www.math.uni-wuppertal.de/~buhl

DATUM 2. Juli 2014

Formale Methoden

SS 2014 – Übungsblatt 11

Ausgabe: 2. Juli 2014

Abgabe bis 9. Juli 2014 an: dsavvidi+fm@studs.math.uni-wuppertal.de

Aufgabe 1. *OCL-Constraints von Person (Personenstandsdaten)*

Realisieren Sie die OCL-Constraints aus Abschnitt 2.13 der Vorlesung mit Hilfe von Papyrus. Benutzen Sie notfalls die schon diskutierten Workarounds! (Wo sind solche nötig?)

Aufgabe 2. *OCL-Constraints*

Konzipieren Sie einen Aufzählungstyp für Studierende, der Studierende als Gasthörer, als Seniorenstudierende beziehungsweise als Vollzeitstudierende ausweist (Zeichnung eines UML-Klassendiagramms).

Konzipieren Sie die folgenden OCL-Constraints:

- Wenn ein Studierender Gasthörer an einer Universität ist, muss er an einer anderen als Vollzeitstudierender immatrikuliert sein.
- Seniorenstudierende können (spezielle) Teilnahme-Scheine bekommen, alle anderen Studierenden müssen Leistungs-Scheine erwerben. (Konzipieren Sie dazu Methoden `get_Teilnahmeschein()`, ... mit geeigneten Vorbedingungen.)
- Seniorenstudierende müssen mindestens 60 Jahre alt sein.
- Vollzeitstudierende müssen mindestens 12 Jahre alt sein.

Aufgabe 3. *Person::trageEheEin()*

Ergänzen Sie die Fallstudie Personenstandsdaten um die Klasse Standesamt mit einer Methode `trageEheEin()`. Überprüfen Sie mittels OCL-Constraints die Voraussetzungen für die Eheerlaubnis: Werte von Status, Geschlecht, ... der Ehepartner: Die Operation soll den Ergebniswert `True` liefern, falls die Ehe eingetragen werden kann (dann auch Änderungen der Hochzeitsassoziationen, des Attributs Status der Ehepartner, ..., in den Nachbedingungen spezifizieren).

Aufgabe 4. *Wochentag()*

Spezifizieren Sie die Nachbedingung der (neu eingeführten) Methode `Wochentag()` der Klasse `Datum` mit Hilfe von `chronoJD(in d : Datum) : chronoJD`.

Aufgabe 5. *Rogério de Lemos: Object constraint language*

Testen Sie die OCL-Constraints aus

Rogério de Lemos: Object constraint language (OCL),

(https://www.cs.kent.ac.uk/teaching/09/modules/CO/8/86/rdl/ocl_4Slides.pdf)

in Papyrus.