

# Übungen zu Softwareentwicklung III, Funktionale Programmierung

Blatt 11, Woche 13

Relationalen Programmierung

Leonie Dreschler-Fischer

WS 2015/2016

**Ausgabe:** Freitag, 8.1.2016,

**Abgabe der Lösungen:** bis Montag, 18.1.2016, 12:00 Uhr per email bei den Übungsgruppenleitern.

**Ziel:** Mit den Aufgaben auf diesem Übungsblatt üben Sie die Verwendung von imperativen Sprachelementen in Racket und lernen einige typische Aspekte der relationalen Programmierung kennen.

**Bearbeitungsdauer:** Die Bearbeitung sollte insgesamt nicht länger als 3 1/2 Stunden dauern.

**Homepage:**

[http://kogs-www.informatik.uni-hamburg.de/~dreschle/teaching/  
Uebungen\\_Se\\_III/Uebungen\\_Se\\_III.html](http://kogs-www.informatik.uni-hamburg.de/~dreschle/teaching/Uebungen_Se_III/Uebungen_Se_III.html)

Bitte denken Sie daran, auf den von Ihnen eingereichten Lösungsvorschlägen *Ihren Namen und die Matrikelnummer, den Namen der Übungsgruppenleiterin / des Übungsgruppenleiters und Wochentag und Uhrzeit der Übungsgruppe* anzugeben, damit wir ihre Ausarbeitungen eindeutig zuordnen können.

# 1 Prolog in Racket

## 1.1 Unifikation

Bearbeitungszeit: 1 Std.,

10 Pnkt.

Unifizieren Sie die folgenden Ausdrücke (falls möglich) und geben Sie für den Fall, dass die Unifikation erfolgreich ist, die dabei erzeugten Variablenbindungen an.

	Ausdruck 1	Ausdruck 2
a)	(Laptop ?Marke schwarz)	(Laptop MacBook ?Farbe)
b)	(Teilnehmer ( (Peter Lustig Gold) (Petra Pfiffig Gold)))	(Teilnehmer ( (Peter Lustig Gold) (Petra Pfiffig Silber)))
c)	(Teilnehmer ( (Peter Lustig Gold) (Petra Pfiffig Gold)))	(Teilnehmer ( (Peter Lustig Gold) (Petra Pfiffig ?Rang)))
d)	(Teilnehmer ( (Peter Lustig Gold) . ?andere))	(Teilnehmer ( (Peter Lustig Gold) (Petra Pfiffig Gold) (Lena Lustig Silber)))
e)	(Tupel (k ?farbe As ) (k Pik ?wert ) )	(Tupel (k Pik ?wert) (k ?farbe As) )
f)	(Tupel (k ?farbe As ) (k Pik ?wert2 ) )	(Tupel (k Pik ?wert) (k ?farbe ?wert) )

## 1.2 Anfragen

Bearbeitungszeit: 1 Std.,

10 Pnkt.

Eine relationale Datenbank für Bibliotheksausleihen sei als Faktensammlung in Prolog-in-Racket implementiert und enthalte die Relationen

- ([ausleihe](#) Signatur Lesernummer) für jedes derzeit ausgeliehene Buch,
- ([vorbestellung](#) Signatur Lesernummer) für jedes derzeit vorbestellte Buch, und
- ([leser](#) Name Vorname Lesernummer Geburtsjahr) für die persönlichen Angaben jedes Lesers.

```
;(ausleihe Signatur Lesernummer)
(<- (ausleihe "P_201" 100))
```

```

(<- (ausleihe "P_30" 102))
(<- (ausleihe "P_32" 104))
(<- (ausleihe "P_50" 104))
;( vorbestellung Signatur Lesernummer)
(<- (vorbestellung "P_201" 104))
(<- (vorbestellung "P_201" 102))
(<- (vorbestellung "P_30" 100))
(<- (vorbestellung "P_30" 104))
;( leser Name Vorname Lesernummer Geburtsjahr)
(<- (leser Neugierig Nena 100 1989))
(<- (leser Linux Leo 102 1990))
(<- (leser Luator Eva 104 1988))

```

Stellen Sie die folgenden Anfragen an die Datenbank. Formulieren Sie ihre Anfragen so, dass die Systemausgabe weitgehend selbsterklärend ist:

1. Ist das Buch mit der Signatur P 201 vorbestellt?
2. Welche Lesernummer hat Leo Linux?
3. Welcher Leser (identifiziert durch Name und Vorname) hat das Buch mit der Signatur P 30 vorbestellt?
4. Welche Leser (identifiziert durch Namen und Vornamen), die älter als 60 Jahre sind, haben ein Buch ausgeliehen?
5. Welche Leser haben mehr als ein Buch ausgeliehen?

## 2 Stromorientierte Programmierung

Diese Aufgabe ist erst nach der Vorlesung am nächsten Freitag zu bearbeiten.  
Abgabe 25.1.2015

Bearbeitungszeit: 1 Std.,

10 Pnkt.

Beim Abzählspiel „Die Böse Sieben“ wird reihum gezählt, aber wenn eine Zahl, die durch Sieben teilbar ist oder die die Ziffer Sieben enthält, genannt werden soll, wird stattdessen „sum“ gesagt: 1, 2, 3, 4, 5, 6, sum, 8, 9, 10, 11, 12, 13, sum, 15, 16, sum, 18 ...

Definieren Sie eine Stromfunktion in der Sprache Lazy, die den Strom der natürlichen Zahlen erzeugt, aber alle durch Sieben teilbaren Zahlen und Zahlen mit der Ziffer Sieben durch das Symbol „sum“ ersetzt.

### 3 Memo-Funktionen

Diese Aufgabe ist erst nach der Vorlesung am nächsten Freitag zu bearbeiten.  
Abgabe 25.1.2015

Bearbeitungszeit: 30 Minuten,

5 Pnkt.

Definieren Sie eine rekursive Funktion zur Berechnung der Fakultät einer Zahl als Memo-Funktion, die sich schon berechnete Werte in einer Tabelle merkt. Stellen Sie sicher, daß auch die rekursiven Aufrufe an die Memo-Funktion gehen. Zur Memoisierung können Sie die Funktion memo aus dem tools-module verwenden.

**Erreichbare Punkte:** 35

**Erreichbare Zusatzunkte:** 0