Übungen zu Softwareentwicklung III, Funktionale Programmierung

Blatt 9, Woche 11

Funktionen höherer Ordnung und kombinatorische Probleme

Leonie Dreschler-Fischer

WS 2012/2013

Ausgabe: Freitag, 4.1.2012,

Abgabe der Lösungen: bis Montag, 14.1.2012, 12:00 Uhr per email bei den Übungsgruppenleitern.

Ziel: Die Aufgaben auf diesem Zettel dienen dazu, sich mit dem Entwurf von Funktionen höherer Ordnung und Rekursion zur Lösung kombinatorischer Probleme vertraut zu machen.

Bearbeitungsdauer: Die Bearbeitung sollte insgesamt nicht länger als 4 Stunden dauern.

Homepage:
`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 Listen als Mengen

(Bearbeitungszeit 2 1/2 Std.)

1.1 Allquantor und Existenzquantor als Funktionen höherer Ordnung:

6 Pnkt.

1. Definieren Sie ein Prädikat `every p? xs`, das überprüft, ob für *jedes* Element der Liste xs das Prädikat p? erfüllt ist. `every` entspricht dem *Allquantor„für alle“*.

   `(every (curry = 3) '(3 3 3))` $\longrightarrow$ #**t**

2. Definieren Sie weiterhin ein Semiprädikat `some p? xs`, das überprüft, ob mindestens ein Element der Liste xs das Prädikat p? erfüllt und das gefundene Element als Resultat zurückgibt.

   `some` entspricht dem Existenzquantor *„es gibt“*.

   `(some (curry = 3) '(1 3 9) )` $\longrightarrow$ 3

1.2 Prädikate über Relationen

14 Pnkt.

Gegeben sie eine Menge m, repräsentiert als Liste, sowie eine Relation r als Teilmenge von $m \times m$, die als Liste von *dotted pairs* repräsentiert sei. Verwenden Sie die Prädikate **some** und **every**, um die folgenden Prädikate für Relationen zu definieren:

symmetrisch?: Ist r symmetrisch?

asymmetrisch?: Ist r asymmetrisch?

reflexiv?: Ist r reflexiv?

transitiv? Ist r transitiv?

aequi? r: Ist r eine Äquivalenzrelation?

ord? r: Ist r eine strikte Ordnungsrelation?

2 Das Kreuzprodukt von Mengen: Baumrekursion

(Bearbeitungszeit 1. 1/2 Std.), 20 Pnkt.

Gegeben seien Mengen $M_1, M_2, \ldots$ als Listen ms1, ms2, ... repräsentiert.

1. Definieren Sie eine Funktion Kreuzprodukt, die die Menge $M_1 \times M_2$ errechnet.

   Repräsentieren Sie die Tupel der Relation als Listen mit zwei Elementen. (m1 m2). Beispiel:

$$
\begin{aligned}
M_1 &= \{1, 2\} \\
M_2 &= \{a, b, c\} \\
M_1 \times M_2 &= \{(1, a)(1, b)(1, c)(2, a)(2, b)(2, c)\}
\end{aligned}
$$

```
(Kreuzprodukt '(1 2) '(a b c))
  ⟶ '( (1 a) (1 b) (1 c) (2 a) (2 b) (2 c))
```

2. Verallgemeinern Sie die Funktion, so daß Sie das Kreuzprodukt beliebig vieler Mengen berechnen können.

   Die Menge der Basismengen sei als Liste von Listen repräsentiert.

```
(Produkt '((1 2) (a b) (X)))
   ⟶ '( (1 a X) (1 b X) (2 a X) (2 b X) (2 c X))
```

3. Definieren Sie eine Funktion Kombination, die die Liste aller Kombinationen (Auswahl von k verschiedenen Elementen aus einer Menge M der Mächtigkeit n ohne Beachtung der Reihenfolge) berechnet.

```
(Kombination 2 '(a b c) )
   ⟶ '( (a b) (a c) (b c))
```

Erreichbare Punkte: 40

Erreichbare Zusatzunkte: 0