8. Aufgabenblatt vom Samstag, den 14. Dezember 2019 zur Vorlesung

ALP I: Funktionale Programmierung

Bearbeiter: A. Rudolph und F. Formanek
Tutor: Stephanie Hoffmann
Tutorium 06

Abgabe: bis Montag, den 06. Januar 2020, 10:10 Uhr

1. **Aufgabe** (24 Punkte)

(a) *Behauptung: reverse(reverse xs) = xs*

Induktionsanfang: xs = [ ]
reverse(reverse [ ]) $\overset{rev.1}{=}$ reverse [ ] $\overset{rev.1}{=}$ [ ]

Induktionsvorraussetzung: für xs = xs' gilt:
reverse(reverse xs') = xs'

Indukionsschritt: Sei xs = (x:xs')
reverse(reverse (x:xs')) $\overset{rev.2}{=}$ reverse(reverse xs' ++ [x]) $\equiv$
(reverse [x]) ++ reverse (reverse xs') $\overset{rev.2}{=}$
(reverse ([ ]) ++ [x]) ++ reverse(reverse xs') $\overset{rev.1}{=}$
([ ] ++ [x]) ++ reverse(reverse xs') $\overset{(++).1}{=}$
[x] ++ reverse(reverse xs') $\overset{nachIV}{=}$
[x] ++ xs' $\equiv$ (x:xs')

Das bedeutet, dass die Behauptung für alle xs (endliche Listen) gilt.

(b) *Behauptung: reverse(xs ++ ys) = reverse ys ++ reverse xs*

Induktionsanfang: xs = [ ]
reverse([] ++ ys) = reverse ys ++ reverse [] $\overset{rev.1}{=}$
reverse([] ++ ys) = reverse ys ++ [] $\overset{(++).1}{=}$
reverse ys = reverse ys

Induktionsvorraussetzung: für xs = xs' gilt:
reverse(xs' ++ ys) = reverse ys ++ reverse xs'

Indukionsschritt: Sei xs = (x:xs')
reverse((x:xs') ++ ys) = reverse ys ++ reverse(x:xs') $\overset{rev.2}{=}$
reverse((x:xs') ++ ys) = reverse ys ++ (reverse xs' ++ [x]) $\overset{(++).2}{=}$

reverse(x:(xs'++ys)) = reverse ys ++ (reverse xs' ++ [x]) $\stackrel{rev.2}{=}$
reverse(xs'++ys) ++ [x] = reverse ys ++ (reverse xs' ++[x]) $\stackrel{nach IV}{=}$
reverse ys ++ reverse xs' ++ [x] = reverse ys ++ (reverse xs' ++ [x]) $\equiv$
reverse ys ++ reverse xs' ++ [x] = reverse ys ++ reverse xs' ++ [x]

Das bedeutet, dass die Behauptung für alle xs (endliche Listen) gilt.