1 Übung 4

1.1

```
reverse::List a->List a
reverse Nil = Nil
reverse (Cons x xs) = snoc (reverse xs) x
```

$$\forall xs\ ys.reverse(xs \oplus ys) = (reverse\ ys) \oplus (reverse\ xs)$$

Sei ys::List a beliebig aber fest.(so schwachsinnige diese schreibweise auch ist)

I.A. xs =Nil

wir erhalten¸

reverse (Nil⊕ ys) = reverse ys

und

(reverse ys)⊕(reverse Nil) = (reverse ys)⊕Nil = reverse ys

per Übung 2.3.a

Der induktionsanfang gilt also für beliebige ys.

I.V.: xs=Cons z zs, für zs gilt

reverse (zs⊕ ys)= (reverse ys)⊕(reverse zs)

linke Seite Umformen:

reverse ((Cons z zs)⊕ ys) $\overset{def\ \oplus}{=}$ reverse (Cons z (zs⊕ys)) $\overset{def\ reverse}{=}$ snoc (reverse (zs⊕ys)) z $\overset{def\ IV}{=}$

snoc ((reverse ys)⊕(reverse zs)) z

von der anderen Seite

(reverse ys)⊕(reverse (Cons z zs)) $\overset{def\ reverse}{=}$ (reverse ys)⊕(snoc (reverse zs) z) $\overset{(1)}{=}$ snoc ((reverse ys)⊕(reverse zs)) z

(1) Anwendung von 2.3.b von rechts nach links.

Somit gilt auch die I.V. also gilt die Aussage.

1.2

1.2.1

$$\forall xs\ ys.\ reverse'\ xs\ ys = (reverse'\ xs\ Nil) \oplus ys$$

(idee: der rechte parameter bleibt unverändert, bis auf hinzufügen der Liste im “stackformat”)

I.A. die Aussage gilt für xs=Nil

reverse’ Nil ys= ys $\overset{def\ \oplus}{=}$Nil ⊕ys= reverse (Nil Nil) ⊕ys per definition von reverse'.

I.V. xs = Cons z zs

Die Aussage gilt für zs: reverse' zs bel=(reverse' zs Nil)⊕bel

reverse' (Cons z zs) ys $\overset{def\ reverse'}{=}$ reverse' zs (Cons z ys) $\overset{IV}{=}$ (reverse' zs Nil) ⊕ (Cons z ys)

andere Seite

(reverse' (Cons z zs) Nil)⊕ ys $\overset{def\ reverse'}{=}$ (reverse' zs (Cons z Nil))⊕ ys $\overset{IV}{=}$ ((reverse' zs Nil) ⊕ (Cons z Nil))⊕ ys $\overset{(1)}{=}$

(reverse' zs Nil) ⊕((Cons z Nil)⊕ ys) $\overset{def\ \oplus}{=}$ (reverse' zs Nil) ⊕(Cons z (Nil⊕ ys)) $\overset{def\ \oplus}{=}$ (reverse' zs Nil) ⊕(Cons z ys)

wobei in (1) die Assoziativität der konkatenation benutzt wurde (Beweis aus Vorlesung cc und ⊕ sind α-Äquivalent)

1.2.2

I.A.: sei xs =Nil

reverse Nil =Nil $\overset{def\ reverse'}{=}$ (reverse' Nil Nil)

gilt.

I.V. xs = Cons z zs

Die Aussage gilt für zs: reverse zs = reverse' zs Nil

reverse (Cons z zs) $\overset{def\ reverse}{=}$ snoc (reverse zs) z $\overset{IV}{=}$ snoc (reverse' zs Nil) z

reverse' (Cons z zs) Nil $\overset{def\ reverse'}{=}$ reverse' zs (Cons z Nil) $\overset{Lemma\ a)}{=}$ (reverse' zs Nil) ⊕ (Cons z Nil) $\overset{3.3}{=}$

snoc (reverse' zs Nil) z

Somit gilt die Aussage.

2

2.1

$$\forall f\ g\ xs.(map\ f.map\ g)\ xs = map\ (f.g)\ xs$$

Sei f,g beliebig, aber fest.

I.A.: xs=Nil

(map f.map g) Nil = $\lambda x.(map\ f)((map\ g)(x))$ Nil = (map f)((map g)(Nil)) = (map f)(Nil) = Nil

Anwenden der definition von (.) und dann 2 mal von map

map (f.g) Nil = Nil.

definition von map.

I.V. xs = Cons z zs

Die Aussage gilt für zs: (map f.map g) zs = map (f.g) zs

(map f.map g) (Cons z zs) $\overset{def\ (.)}{=}$ $\lambda x.(map\ f)((map\ g)(x))$ (Cons z zs) $\overset{applikation}{=}$ (map f)((map g)(Cons z zs)) $\overset{def\ map}{=}$

(map f)(Cons (g z) (map g zs)) $\overset{def\_map}{=}$ Cons (f (g z)) (map f (map g zs))

Andere Seite

map (f.g) (Cons z zs) $\overset{def\_map}{=}$ Cons ((f.g) z) (map (f.g) zs) $\overset{def\_(.)}{=}$ Cons (($\lambda x.f(g\ x)$) z) (map (f.g) zs) $\overset{\beta}{=}$

Cons (f(g z)) (map (f.g) zs) $\overset{IV}{=}$ Cons (f(g z)) ((map f.map g) zs) $\overset{def\_(.)}{=}$ Cons (f(g z)) ($\lambda x.(map\ f)((map\ g)(x))$ zs) $\overset{\beta}{=}$

Cons (f(g z)) ((map f)((map g)(zs))) $\overset{applikation}{=}$ Cons (f(g z)) ((map f)((map g zs)) $\overset{applikation}{=}$

Cons (f(g z)) (map f (map g zs))

Somit gilt die Aussage.

2.1.1

$$\forall f\ ysxs.map\ f\ (xs \oplus ys) = (map\ f\ xs) \oplus (map\ f\ ys)$$

I.A. xs=Nil

map f ($Nil \oplus ys$) $\overset{def\_\oplus}{=}$ map f ys

(map f Nil) ⊕ (map f ys) $\overset{def\_map}{=}$ Nil ⊕ map f ys $\overset{def\_oplus}{=}$ map f ys

I.V. xs = Cons z zs

Die Aussage gilt für zs: map f (zs⊕ ys) = (map f zs) ⊕ (map f ys)

map f ((Cons z zs)⊕ ys) $\overset{def\_\oplus}{=}$ map f ((Cons z (zs⊕ ys) $\overset{def\_map}{=}$ Cons (f z) (map (zs⊕ ys)) $\overset{IV}{=}$

Cons (f z) ((map f zs) ⊕ (map f ys))

Andere Seite.

(map f (Cons z zs)) ⊕ (map f ys) $\overset{def\_map}{=}$

(Cons (f z) (map f zs))) ⊕ (map f ys) $\overset{def\_\oplus}{=}$ Cons (f z) ((map f zs) ⊕ (map f ys))

Aussage gilt.

3

3.1

$$\forall t.mirror\ (mirror\ t) = t$$

I.A. t=Leaf.

mirror (mirror (Leaf)) $\overset{def\ mirror}{=}$ mirror (Leaf) $\overset{def\ mirror}{=}$ Leaf =Leaf

gilt.

I.V. t = Bin left x right

Die Aussage gilt für left und right.

mirror (mirror left) = left

mirror (mirror right) = right

mirror (mirror (Bin left x right)) $\overset{def\ mirror}{=}$ mirror (Bin (mirror right) x (mirror left)) $\overset{2\times IV}{=}$ mirror (Bin right x left) $\overset{def\ mirror}{=}$ (Bin (mirror left) x (mirror right) $\overset{IV}{=}$ Bin left x right

3.2

$$\forall t. inorder\ (mirror\ t) = reverse\ (inorder\ t)$$

I.A. t = Leaf

inorder (mirror (Leaf)) $\overset{def\ mirror}{=}$ inorder (Leaf) $\overset{def\ inorder}{=}$ Nil

andere Seite:

reverse (inorder Leaf) $\overset{def\ indrder}{=}$ reverse Nil $\overset{def\ reverse}{=}$ Nil

gilt.

I.V. t = Bin left x right

Die Aussage gilt für left und right.

inorder (mirror left) = reverse (inorder left)

inorder (mirror right) = reverse (inorder right)

inorder (mirror (Bin left x right)) $\overset{def\ mirror}{=}$ inorder (Bin (mirror right) x (mirror left)) $\overset{def\ inorder}{=}$ inorder (mirror right) ⊕ (Cons x (inorder (mirror left))) $\overset{2\times IV}{=}$ reverse (inorder right) ⊕ (Cons x (reverse (inorder left))

Andere Seite:

reverse (inorder (Bin left x right)) $\overset{def\ inorder}{=}$ reverse (inorder left ⊕ (Cons x (inorder right))) $\overset{4.1}{=}$ (reverse (Cons x (inorder right)))) ⊕ reverse (inorder left) $\overset{def\ reverse}{=}$ (snoc (reverse (inorder right)) x) ⊕ reverse (inorder left) $\overset{Lemma\ A}{=}$ reverse (inorder right) ⊕ (Cons x (reverse (inorder left))

Die Aussage gilt.

<u>Lemma A:</u>

(snoc xs x) ⊕ ys = xs ⊕ (Cons x ys)

I.A. xs = Nil

(snoc Nil x) ⊕ ys $\overset{def\ snoc}{=}$ (Cons x Nil) ⊕ ys $\overset{def\ \oplus}{=}$ Cons x (Nil⊕ ys) $\overset{def\ \oplus}{=}$ Cons x ys

andere Seite:

Nil ⊕ (Cons x ys) $\overset{def\ \oplus}{=}$ Cons x ys.

gilt.

I.V. xs=Cons z zs

Die Aussage gilt für zs: (snoc zs x) ⊕ ys = zs ⊕ (Cons x ys).

(snoc (Cons z zs) x) ⊕ ys $\overset{def\ snoc}{=}$ (Cons z (snoc zs x))⊕ ys $\overset{def\ \oplus}{=}$ Cons z ((snoc zs x)⊕ ys) $\overset{IV}{=}$ Cons z (zs ⊕ (Cons x ys))

Andere Seite:

(Cons z zs) ⊕ (Cons x ys) $\overset{def\ \oplus}{=}$ Cons z (zs⊕(Cons x ys)).

Das Lemma ist gültig.