

Informatik I

PD Dr. J.-G. Smaus
G. Röger, R. Mattmüller
Wintersemester 2010/2011

Universität Freiburg
Institut für Informatik

Übungsblatt 5

Abgabe: 13. Januar 2011

Sprachlevel: Die Macht der Abstraktion – fortgeschritten.

Aufgabe 5.1 (Türme von Hanoi, 3+2+2 Punkte, Abg.: `hanoi.rkt`, auf Papier)

- (a) Schreiben Sie die in der Vorlesung vorgestellte Implementierung der Lösung des Problems der Türme von Hanoi¹ so um, dass die `hanoi`-Funktion neben der Anzahl der zu verschiebenden Scheiben auch die Nummern des Start-, des Hilfs- und des Zielpfahls als Argumente bekommt.
- (b) Sei $f(n)$ die Anzahl der durch das Programm aus (a) berechneten Züge bei n Scheiben. Geben Sie eine (nicht-rekursive) Formel für $f(n)$ an und beweisen Sie die Korrektheit der Formel durch Induktion über n .
Hinweis: Hilft es, zuerst eine Rekursionsgleichung aufzustellen?
- (c) Sei $f^*(n)$ die Länge einer *kürzesten* Zugfolge für das Problem der Türme von Hanoi mit n Scheiben. Beweisen Sie, dass $f^*(n) = f(n)$ für alle $n \in \mathbb{N}$.

Aufgabe 5.2 (Fkt. höherer Ordnung, 2+2 Punkte, Abg.: `first-with.rkt`)

- (a) Schreiben Sie eine Funktion `(: first-with ((%a -> boolean) (list %a) -> (mixed %a (one-of #f))))`, so dass `(first-with p xs)` das erste x in xs liefert, für das $p(x)$ wahr ist, und `#f`, wenn kein solches x in xs existiert. Die Funktion `first-with` soll *ohne* Verwendung von `filter` auskommen.
- (b) Schreiben Sie eine Funktion `first-with-using-filter` mit der gleichen Ein- und Ausgabe wie `first-with`, jetzt aber unter Verwendung der Definition von `filter` aus der Vorlesung (`filter` können Sie auch mit Hilfe des eingebauten `fold` statt `list-fold` definieren).

Aufgabe 5.3 (Fkt. höherer Ordnung, 1+1+1 Punkte, Abg.: `filter.rkt`)

Schreiben Sie folgende Funktionen unter Verwendung der Definition von `filter` aus der Vorlesung (wie oben können Sie `filter` auch mit Hilfe des eingebauten `fold` statt `list-fold` definieren):

- (a) Eine Funktion `(: evens ((list natural) -> (list natural)))` derart, dass `(evens xs)` die Liste aller *geraden* Zahlen aus xs in der ursprünglichen Reihenfolge zurückgibt.

¹<http://gki.informatik.uni-freiburg.de/teaching/ws1011/infoI/scheme/hanoi.rkt>

- (b) Eine Funktion (`: count-zeroes ((list natural) -> natural)`), die in einer Liste von Zahlen die Nullen zählt.
- (c) Eine Funktion mit Signatur (`: multiples (natural (list natural) -> (list natural))`), so dass (`multiples n xs`) die Liste aller Zahlen aus `xs`, die Vielfache von `n` sind, in der ursprünglichen Reihenfolge zurückgibt.

Aufgabe 5.4 (Fkt. höherer Ordnung, 1+1 Punkte, Abg.: `any.rkt`)

- (a) Schreiben Sie eine Funktion (`: any? ((%a -> boolean) (list %a) -> boolean)`) analog zu `every?` (vgl. Vorlesung), die genau dann `#t` zurückgibt, wenn mindestens ein Element der Liste das Prädikat erfüllt, und `#f` sonst.
- (b) Schreiben Sie eine Funktion `any?-using-every?` mit gleicher Ein- und Ausgabe wie `any?`, die einfach `every?` aufruft und selbst die gegebene Liste nicht durchläuft.

Aufgabe 5.5 (Schönfinkel-Isomorphismus, 2+2 Punkte, Abg.: auf Papier)

Sei p_1 eine Funktion mit einem Parameter, die einparametrische Funktionen zurückgibt, und p_2 eine Funktion mit zwei Parametern. Zeigen Sie:

- (a) $(\text{curry } (\text{uncurry } p_1)) \equiv p_1$
Hinweis: Zeigen Sie dazu, dass für passende Parameter x und y gilt:
 $((\text{curry } (\text{uncurry } p_1)) x) y = (p_1 x) y$.
- (b) $(\text{uncurry } (\text{curry } p_2)) \equiv p_2$
Hinweis: Zeigen Sie dazu, dass für passende Parameter x und y gilt:
 $(\text{uncurry } (\text{curry } p_2)) x y = (p_2 x y)$.

Um für die Programmieraufgaben Punkte zu erhalten, folgen Sie den Konstruktionsanleitungen der Vorlesung, d. h.:

1. Geben Sie die Signatur an.
Falls die Signatur fehlt, gibt die Aufgabe 0 Punkte.
2. Wählen Sie abhängig von der Signatur das richtige Funktionsgerüst aus.
3. Geben Sie pro Funktion mindestens zwei Testfälle an.
4. Schreiben Sie den Funktionsrumpf. Dieser Schritt gliedert sich in weitere Unterschritte, die Sie entsprechend auswählen müssen.
Sie erhalten für diesen Teil der Aufgabe nur Punkte, wenn Sie Schritte 1 bis 3 befolgt haben.

Die Übungsblätter müssen individuell bearbeitet werden. Gruppenabgaben sind nicht zulässig.