

Programmierung 1 (Wintersemester 2012/13)

Aufgaben aus den Übungsgruppen 14 (Kapitel 15 und 16)

Hinweis: Dieses Übungsblatt enthält von den Tutoren für die Übungsgruppe erstellte Aufgaben.

Die Aufgaben und die damit abgedeckten Themenbereiche sind für die Klausur weder relevant, noch irrelevant.

Aufgabe 14.1 (*Lineare Darstellung von Listen*)

Sie haben in Kapitel 15.10 eine Möglichkeit kennen gelernt, um Listen von Zahlen in einer Halde darstellen zu können. Auf diese können Sie die schon bekannten Prozeduren auf Listen aus Kapitel 4 anwenden.

Hinweis: Allokieren Sie außer für *tabulate* keinen neuen Speicherplatz, wenn dies nicht unbedingt notwendig ist !

- Schreiben Sie eine Prozedur $length : address \rightarrow int$, die zu einer angegebenen Adresse die Länge der angegebenen verketteten Liste bestimmt!

`fun length a =`

- Schreiben Sie eine Prozedur $map : address \rightarrow (int \rightarrow int) \rightarrow int$, die auf die Liste einer angegebenen Adresse die Prozedur *map* anwendet!

`fun map a f =`

- Schreiben Sie eine Prozedur $exists : address \rightarrow (int \rightarrow bool) \rightarrow bool$, die auf die Liste einer angegebenen Adresse die Prozedur *filter* anwendet!

`fun exists f a =`

- Schreiben Sie eine Prozedur $all : address \rightarrow (int \rightarrow bool) \rightarrow bool$, die auf die Liste einer angegebenen Adresse die Prozedur *all* anwendet!

`fun all f a =`

- Schreiben Sie eine Prozedur $filter : address \rightarrow (int \rightarrow bool) \rightarrow int$, die auf die Liste einer angegebenen Adresse die Prozedur *filter* anwendet!

Tipp: Sie benötigen eine Hilfsprozedur, um sich das letzte gültige Argument merken zu können!

`fun filter f a =`

- Schreiben Sie eine Prozedur $tabulate : (int * int \rightarrow int) \rightarrow address$, die für eine Zahl *n* und eine Prozedur *tabulate* anwendet!

`fun tabulate (n,f) =`

- Schreiben Sie eine Prozedur $rev : address \rightarrow address$, die die Liste einer Adresse reversiert!

`fun rev a =`