



# Universität Karlsruhe (TH)

## Lehrstuhl für Programmierparadigmen

Theorembeweiser und ihre Anwendungen SS 2009 <http://pp.info.uni-karlsruhe.de/>

Übungsleiter: Daniel Wasserrab

[wasserra@ipd.info.uni-karlsruhe.de](mailto:wasserra@ipd.info.uni-karlsruhe.de)

Übungsblatt 7

Besprechung: 09.06.2009

## 1. Listen zusammenfügen

In dieser Aufgabe soll eine Funktion *zip* definiert werden, welche zwei Listen durch Verschachtelung zusammenfügt. Beispiel:  $zip [a1, a2, a3] [b1, b2, b3] = [a1, b1, a2, b2, a3, b3]$  und  $zip [a1] [b1, b2, b3] = [a1, b1, b2, b3]$ . Verwenden Sie dazu die totale Rekursion.

Zeigen Sie, dass *zip* distributiv über *append* (also *@*) ist.

## 2. Merge Sort

Wir arbeiten im Folgenden nur auf Listen über natürlichen Zahlen.

Definieren Sie ein Prädikat *sorted*, welches prüft, ob jedes Element kleiner oder gleich den folgenden ist;  $le\ n\ xs$  ist *True* g.d.w. *n* kleiner oder gleich allen Elementen in *xs*.

**consts**

```
le      :: "nat ⇒ nat list ⇒ bool"
sorted :: "nat list ⇒ bool"
```

Implementieren Sie nun *Merge Sort*: Eine Liste wird durch Aufteilung in zwei Listen sortiert, welche einzeln sortiert und wieder zusammengefügt werden.

Definieren Sie mittels *fun* zwei Funktionen

```
consts merge :: "nat list ⇒ nat list ⇒ nat list"
        msort :: "nat list ⇒ nat list"
```

und zeigen Sie

```
theorem "sorted (msort xs)"
oops
```

Sie werden dafür Hilfslemmas über *le* und *sorted* beweisen müssen.

Hinweise:

- Um eine Liste in zwei fast gleichlange Hälften zu zerteilen, können Sie die Funktionen *n div 2*, *take* und *drop* verwenden, wobei *take n xs* die ersten *n* Elemente von *xs* zurückgibt, *drop n xs* den Rest.
- Versuchen Sie erstmal, das Lemma alleine zu lösen und selbst herauszufinden, welche Hilfslemmas Sie dafür brauchen. Falls Sie so nicht weiterkommen, hier ein paar Überlegungen:
  - wie verhält sich *le*, wenn der erste Parameter kleiner wird?
  - was muss gelten, wenn der zweite Parameter von *le merge* ist?
  - was muss gelten, damit *merge sorted* ist?