Prof. Dr. Manfred Schmidt-Schauß
Künstliche Intelligenz/Softwaretechnologie

Fachbereich Informatik (20)
Johann Wolfgang Goethe-Universität Frankfurt am Main

Funktionale Programmierung I

Sommersemester 2000

Übungsblatt Nr. 5

Abgabe: Dienstag 9. Mai 2000 vor der Vorlesung

Aufgabe 1 (10 Punkte)

Wie kann man sicherstellen, daß in einem Haskell-Programm nur solche Typen verwendet werden dürfen, die ein bestimmtes Element bzw. Konstruktor enthalten?

Geben Sie Ihre Lösung im Bezug auf das ausgezeichnete Blank-Symbol des Bandalphabets einer Turingmaschine an und implementieren Sie sie für mindestens einen Datentypen Ihrer Wahl.

Aufgabe 2 (25 Punkte)

Implementieren Sie in Haskell die Simulation einer Turingmaschine. Benutzen Sie dabei die in der Vorlesung vorgestellten Konzepte zur Verknüpfung der Zustandsmonade mit Ein-/Ausgabe.

1. Definieren Sie die Konfiguration einer Turingmaschine mit Hilfe eines *Record*-Typen.
2. Schreiben Sie eine Funktion, die die einzelnen Zustandsübergänge der TM ausführt und dabei die momentane Konfiguration ausgibt.
3. Geben Sie auch eine Funktion an, die als Eingabe ein Turingmaschinenprogramm sowie das eigentliche Eingabewort bekommt und die Simulation startet.

Verwenden Sie dabei die Charakterisierung des Bandalphabets aus Aufgabe 1, um Turingmaschinen sowohl auf Eingabeworten „10101“ wie auch „abba“, usf. simulieren zu können.

Aufgabe 3 (15 Punkte)

In Aufgabenblatt 3 wurde die fold-Operation auf Bäumen definiert. Bezeichne `ar(C)` die Stelligkeit eines Konstruktors und sei der Datentyp `T a` mit den Konstruktoren `C_1, ..., C_m` gegeben durch

```
data T a = C_1 t_1_1 ... t_1_ar(C1)
         | C_2 t_2_1 ... t_2_ar(C2)
           ...
         | C_m t_n_1 ... t_m_ar(C2)
```

Welchen Typ sollte eine fold-Funktion auf diesem allgemeinen Datentypen haben?