Institut für Informatik
der Universität München
Prof. M. Hofmann
A. Abel

WS 2005/2006

27. Januar 2006

Übungen zur Vorlesung
Rechnergestütztes Beweisen

Blatt 13

Aufgabe 31: [Papier, 6 Punkte] Eine Liste l_0 heisst *Präfix* einer Liste l, falls es eine Liste l_1 gibt, so dass $l = l_0 ++ l_1$. (Dabei ist $++$ die Konkatenation.)

1. Geben Sie eine induktive Definition von *Präfix* an.
2. Beweisen Sie durch Induktion, dass Ihre Definition die Spezifikation erfüllt. (Korrektheit)
3. Beweisen Sie: Falls l_0 Präfix von l, dann stehen die beiden Listen in der induktiven Relation. (Vollständigkeit.)

Aufgabe 32: [Coq, 6 Punkte] Implementieren Sie die vorige Aufgabe in Coq. Verwenden Sie dazu die Bibliothek `List`.

Aufgabe 33: [Coq, 8 Punkte] **Datentypen mit Invarianten, Spezifikationen und Programmextraktion.** In der Typentheorie kann man Datentypinvarianten mit Hilfe des Typsystems sicherstellen. Ein Beispiel sind binäre Suchbäume, die für jeden Knoten folgende Invariante erfüllen: alle Schlüssel im linken Teilbaum sind kleiner als der im Knoten abgelegte Schlüssel, alle Schlüssel im rechten Teilbaum sind größer oder gleich. Auch können Funktionen in ihrem Typ eine volle Spezifikation enthalten. Z.B. eine Suchfunktion auf dem binären Baum, die bei Erfolg zusätzlich einen Beweis zurückgibt, dass sich der Schlüssel im Baum befindet, und bei Misserfolg einen Gegenbeweis.

Auf den Vorlesungsseiten finden Sie eine teilweise Implementierung diese Suchbäume. Füllen Sie die Lücken!

Abgabe: am Mittwoch 01.02. in der Vorlesung. Elektronische Lösungsteile per Mail an `abel@informatik.uni-muenchen.de`.