

# Abstrakte Maschinen

*Sommersemester 2006*

8. Übungsblatt

Abgabetermin: Do, 6. Juli 2006 in der Vorlesung

Aufgabe 1: Bäume

10 Punkte

Erweitern sie PuF um den Typ `Tree`. Trees werden aus `Tree`-Elementen mithilfe der Konstante `LEAF` und dem Konstruktor `NODE` aufgebaut. `NODE` baut einen `Tree`-Wert aus einem beliebigen Wert und zwei `Tree`-Werten. Die Syntax von Ausdrücken  $e$  wird erweitert um:

$$e ::= \dots \mid LEAF \mid NODE(e_1, e_2, e_3) \\ \mid (\mathbf{case} \ e_0 \ \mathbf{of} \ LEAF \rightarrow e_1; \ NODE(info, left, right) \rightarrow e_2)$$

Definieren sie die Code-Erzeugungsfunktionen für die neu eingeführten Ausdrücke. Erweitern sie hierzu die Heap-Objekte um neue Objekte von dem Typ `Tree`!

Aufgabe 2: Tail-Recursion

10 Punkte

Zur Durchführung von Tail-Recursion-Optimierungen bei der Übersetzung eines Ausdrucks  $e$  müssen die letzten Aufruf-Vorkommen in  $e$  bestimmt werden. Das Aufruf-Vorkommen  $l \equiv e' e_0 \dots e_{m-1}$  heißt *letzt* in einem Ausdruck  $e$ , falls die Auswertung von  $l$  den Wert für  $e$  liefern kann. Geben sie ein allgemeines Schema für die Berechnung der letzten Aufruf-Vorkommen in einem Ausdruck  $e$  an!