Christian-Albrechts-Universität zu Kiel
Institut für Informatik und Praktische Mathematik

Prof. Dr. W.-P. de Roever
Martin Steffen

Algorithmen & Datenstrukturen

Wintersemester 2002/2003 | **Serie 5** | 26. November 2002

<u>Thema :</u> Bäume

Ausgabetermin: 26. November 2002

Abgabe: 5. Dezember 2002

Aufgabe 1 (Suchbäume mit Lifo-Eigenschaft) Gefordert seien *binäre Suchbäume* mit folgenden drei Wörterbuchoperationen

1. $insert(key, elem)$,
2. $delete(key)$ und
3. $search(key)$,

wobei das Löschen einen der Knoten mit dem passenden Schlüssel, falls vorhanden, löscht und das Suchen, falls erfolgreich, das zugehörige Datenelement zurückliefert. Die Schlüssel im Baum brauchen dabei nicht eindeutig sein, es kann also mehrere Knoten mit dem selben Schlüssel geben.

Bei mehrfachem Vorkommen von Schlüsseln soll zusätzlich folgende *Disziplin* eingehalten werden: die Suche nach einem Schlüssel fördert immer dasjenige passende Element zu Tage, welches zu diesem Schlüssel *zuletzt* eingefügt wurde und das Löschen betrifft dasselbe Element. Mit anderen Worten, die Elemente zu einem festen Schlüssel unterliegen einer *last-in-first-out*-Disziplin.[1]

Ändern Sie die vorgestellten binären Suchbäume geeignet ab und beschreiben Sie, warum Ihre Baumimplementierung die Lifo-Eigenschaft besitzt.

[1] Der Baum verhält sich also nach außen hin wie eine Hashtabelle mit externer Verkettung. Derartige Strukturen werden beispielsweise dazu verwendet, um *Symboltabellen* zu repräsentieren, d.h., die Assoziation von Identifiern zu Werten und ähnlichem, wobei die Stackeigenschaft für die geschachtelten Gültigkeitsbereiche der Identifier benötigt wird.