

5. Übungsblatt

Ausgabe: 11.05.2010 **Abgabe:** 18.05.2010, vor der Vorlesung

Aufgabe 1: Listen

10 Punkte

Betrachten Sie folgende (rudimentäre) Java-Implementierung der Datenstruktur `List`:

```
public class List {  
    // Klassendefinition  
    class Node {  
        Object data;  
        Node next;  
    }  
  
    // Variablen  
    private Node head;  
    private Node tail;  
  
    // Konstruktor  
    public List() {  
        head = new Node();  
        tail = new Node();  
        head.next = tail;  
        head.data = null;  
        tail.next = null;  
        tail.data = null;  
    }  
}
```

Dabei repräsentieren Objekte der Klasse `Node` die Listenelemente, das `Node`-Objekt `head` das erste Listenelement und das `Node`-Objekt `tail` das letzte Listenelement. Beachten Sie, dass `head` und `tail` stets verschiedene Objekte sein sollen.

Ergänzen Sie die Klasse `List` um Implementierungen für folgende zusätzliche Methoden:

- (a) `public Node first()`: Gibt das erste Listenelement zurück (ohne es zu löschen)
- (b) `public boolean isFirst(Node n)`: Gibt `true` zurück, falls `n` das erste Listenelement ist, sonst `false`
- (c) `public Node after(Node n)`: Gibt den Nachfolger des Listenelements `n` zurück

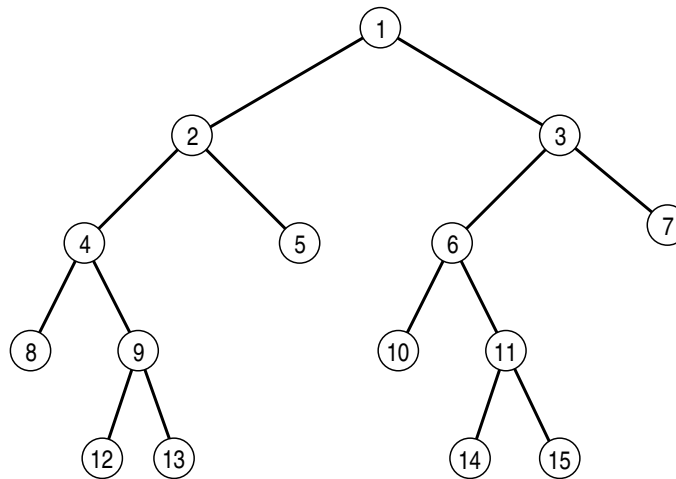
- (d) `public void insertAfter(Node n, Object o)`: Fügt ein neues Listenelement, dessen Inhalt das Objekt `o` ist, als Nachfolger des Elements `n` in die Liste ein
- (e) `public Object removeAfter(Node n)`: Entfernt das dem Listenelement `n` nachfolgende Element aus der Liste und gibt den Inhalt des entfernten Elementes zurück

Aufgabe 2: Traversierung

10 Punkte

Wir betrachten Binärbäume, in deren Knoten Integer-Zahlen als Datenobjekte abgelegt sind. Das auf den Knoten definierte Aktionsobjekt mit der Methode `action(Node n)` gibt stets die im Knoten abgespeicherte Zahl aus.

Geben Sie für den Baum



die Reihenfolgen an, in der die gespeicherten Zahlen bei Präorder-, Inorder- und Postorder-Traversierung ausgegeben werden.

Aufgabe 3: Traversierung

10 Punkte

Wir betrachten Binärbäume, in deren Knoten Integer-Zahlen als Datenobjekte abgelegt sind. Auf den Knoten wird eine Aktion ausgeführt, die die im Knoten gespeicherte Zahl ausgibt.

Geben Sie einen vollen Binärbaum mit 11 Knoten an, in dessen Knoten jede der Zahlen $1, 2, \dots, 11$ genau einmal gespeichert ist, so dass bei der Postorder-Traversierung genau die Folge $(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11)$ ausgegeben wird.