

## Übung 3: Binärbaum

Abgabetermin: 05.04.2011

Name: \_\_\_\_\_ Matrikelnummer: \_\_\_\_\_

Gruppe:  G1 Di 10:15  G2 Di 11:00  G3 Di 12:45

Aufgabe	Punkte	gelöst	abzugeben schriftlich	abzugeben elektronisch	Korr.	Punkte
Aufgabe 1	24	<input type="checkbox"/>	Java-Programm Testfälle und Ergebnisse	Java-Programm	<input type="checkbox"/>	

### Aufgabe 1: Binärer Suchbaum für Integer-Zahlen (24 Punkte)

Implementieren Sie einen binären Suchbaum in der Klasse *BinaryTree*. Der Suchbaum hat folgende Methoden: *insert* fügt eine Zahl sortiert ein, *search* sucht den Knoten mit der gegebenen Zahl, *delete* entfernt eine Zahl und *size* liefert die Anzahl der Zahlen. *preorder*, *inorder* und *postorder* und liefern einen Iterator mit dem der Suchbaum in der jeweiligen Reihenfolge traversiert werden kann.

Implementierungshinweis: Implementieren Sie *insert*, *search* und *size* mit *rekursiven* Algorithmen.

```

class Node {
    int value;
    Node left, right;
}
class BinaryTree {
    Node root;
    void insert(int value) {...}
    Node search(int value) {...}
    Node delete(int value) {...}
    int size() {...}
    TreeIterator preorder() {...}
    TreeIterator inorder() {...}
    TreeIterator postorder() {...}
}
abstract class Iterator {
    public abstract boolean hasNext();
    public abstract int next();
}
class TreeIterator extends Iterator {...}

BinaryTree t = new BinaryTree();
int[] values = new int[] {
    9, 17, 12, 32, 5, 2, 8 };
for (int i=0; i < values.length; i++) {
    t.insert(values[i]);
}
Iterator it = t.inorder();
while (it.hasNext()) {
    Out.print(" " + it.next());
}
// Ausgabe: 2 5 8 9 12 17 32

```

Abzugeben ist: Java-Programm, Testfälle und Ergebnisse