Technische Universität Berlin **SS99**
Fachbereich Mathematik
Sekr. MA 6-1 / Frau Marcus

Dr. Michael Joswig Olaf Jahn
Matthias Müller, André Opitz

4. Programmieraufgabe

Abnahme: spätestens 10.06.99 zur Rechnervorrangzeit

Diese Aufgabe soll den Grundstein für das große Projekt der zweiten Semesterhälfte, die Internet-Suchmaschine, legen. Eine Suchmaschine muß sehr viele Daten speichern und abrufen, so daß wir zunächst dafür eine Datenstruktur vorsehen müssen.

Schreibt also eine Java-Klasse für AVL-Bäume, die das `SimpleMap`-Interface (im wesentlichen Methoden zum Einfügen und Abfragen von Werten) implementiert, das Ihr auf den CoMa-WWW-Seiten findet. Der Baum soll Paare (Schlüssel, Wert) speichern, wobei über den Schlüssel auf den Wert zugegriffen werden können soll.

Seht einen Konstruktor vor, der (nur) einen `Comparer` übergeben bekommt, den Ihr für die Suchbaumeigenschaft benötigt. Ihr müßt von diesem `Comparer` fordern – könnt es jedoch nicht überprüfen –, daß er *verträglich* ist mit der `equals(Object)`-Methode der Schlüsselklasse. (Was bedeutet das und warum ist es wichtig?)

Zum Testen seht irgendwo ein `main` vor, mit dem man (Schlüssel, Wert)-Paare aus einer Datei einlesen und sie anschließend über ihre Schlüssel interaktiv wieder abfragen kann. Auch soll man jetzt noch neue Einträge hinzufügen können. Wie Ihr das genau realisiert, ist Euch überlassen.

Damit Ihr (und wir) die AVL-Eigenschaft überprüfen könnt, sind noch folgende zwei Methoden *erforderlich*:

- die eine bestimmt per Tiefensuche die Höhe eines Teilbaumes unterhalb des übergebenen Knotens, und
- die andere testet mit Hilfe der vorigen Methode, ob alle gespeicherten Balancen stimmen und betragsmäßig höchstens eins sind.

Natürlich sind diese Methoden ausschließlich zur Fehlersuche da. Eure Implementation darf ansonsten *niemals* die Höhe eines Teilbaums per Baumsuche bestimmen, da es viel zu aufwendig ist (was bei der Fehlersuche hingegen nicht stört).

Sicher praktisch ist außerdem eine Methode, die den Baum irgendwie komplett ausgibt. Beispieldateien zum Testen sind vorhanden.