

Übungsblatt 8

Abgabe bis Dienstag, den 18. Juni um 16:00 Uhr

Aufgabe 1 (10 Punkte)

Implementieren Sie einen binären Suchbaum mit Methoden für *insert* und *lookup*, wie in Vorlesung 8a erklärt. Schreiben Sie (wie immer) für beide Methoden einen Unit Test.

Sie brauchen keine verkettete Liste zwischen den Knoten im Baum zu implementieren. Als Schlüsseltyp können Sie *int* annehmen, und die values können Sie für die Aufgabe weglassen.

Zur Spezifikation der Methoden, siehe die Vorlage auf dem Wiki. Dort finden Sie auch bereits eine *toString* Methode (sowohl in Java als auch in C++), die Ihnen beim Testen und Debugging nützlich sein wird.

Aufgabe 2 (10 Punkte)

Zeigen Sie, dass die Kosten von n beliebigen insert / remove Operationen für $(3, 7)$ -Bäume $O(n)$ sind.

Der Beweis ist sehr ähnlich zu dem aus der Vorlesung 8b für $(2, 4)$ -Bäume, es muss lediglich die Potenzialfunktion geeignet abgeändert werden. Achten Sie aber auf eine saubere und vollständige Beweisführung, das heißt, geben Sie nicht nur die geänderte Potenzialfunktion an, sondern zeigen Sie, dass damit die Kosten pro Operation auch tatsächlich in allen Fällen durch die Potenzialdifferenz beschränkt werden können, also $T_i \leq A \cdot (\Phi_i - \Phi_{i-1}) + B$.

Committen Sie Ihren Code (für Aufgabe 1) + Beweis (für Aufgabe 2) in unser SVN, in einen neuen Unterordner *uebungsblatt_08*. Committen Sie in diesem neuen Unterordner außerdem wie gehabt eine Textdatei *erfahrungen.txt*. Beschreiben Sie dort in ein paar Sätzen Ihre Erfahrungen mit diesem Übungsblatt und der Vorlesung dazu. Insbesondere: Wie lange haben Sie ungefähr gebraucht? An welchen Stellen gab es Probleme und wieviel Zeit hat Sie das gekostet?