



Übungsblatt 05

Aufgabe T16

In der Vorlesung wurde für (a, b) -Bäume verlangt, dass $b \geq 2a - 1$ gilt. Daher gibt es $(2, 3)$ -Bäume, aber keine $(3, 4)$ -Bäume.

Warum gilt diese Einschränkung? Was für Probleme könnte es beispielsweise mit $(3, 4)$ -Bäumen geben?

Aufgabe T17

Wir betrachten die Hashfunktion $h: \Sigma^* \rightarrow \{0, \dots, m-1\}$, $c_1 c_2 \dots c_k \mapsto \left(\sum_{i=1}^k c_i \right) \bmod m$, welche die Menge Σ^* aller Wörter (im ASCII-Alphabet) auf eine Zahl zwischen 0 und $m-1$ abbildet.

- Ist h eine gute Hashfunktion? Überlegen Sie sich realistische Anwendungsbeispiele, für welche h sehr viele Kollisionen erzeugt.
- Wie könnte eine vernünftige Hashfunktion für Zeichenketten aussehen, für welche Sie unter normalen Umständen ein gutes Verhalten erwarten würden? Wie gehen Sie mit dem Fall um, dass m sehr groß ist?

Aufgabe T18

Wir verwenden jetzt folgende universelle Familie von Hashfunktionen:

$$\{h_{a,b} \mid 1 \leq a < 5, 0 \leq b < 5\}$$

mit $h_{a,b}(x) = ((ax + b) \bmod 5) \bmod 4$.

Berechnen Sie die Wahrscheinlichkeit, dass $h(2) = h(3)$ ist, falls wir h zufällig aus obiger Familie von Funktionen wählen. Was müsste herauskommen, wenn man bedenkt, dass es sich um eine universelle Familie von Hashfunktionen handelt?

Da es sich um viele Funktionen handelt, sollte die Arbeit unter vielen Personen verteilt werden, damit es schneller geht. Wie lange brauchen Sie gemeinsam, um diese Wahrscheinlichkeit ohne Taschenrechner oder ähnlichem zu berechnen?

Aufgabe T19

Fügen Sie in einen leeren Splaybaum die Schlüssel 5, ..., 10 in dieser Reihenfolge ein. Löschen Sie dann die 7 und anschließend die 8. Fügen Sie danach die 8 wieder ein.

Aufgabe H13 (5+1+3 Punkte)

- Erstellen Sie mithilfe eines Programms eine Tabelle, welche die Wahrscheinlichkeiten von $h(x) = h(y)$ für alle $0 \leq x < 5, 0 \leq y < 5$ enthält, falls h wieder zufällig aus der Familie von T18 gezogen wird.
- Ist das Ergebnis das, was Sie erwarten? Warum?
- Wiederholen Sie das Experiment für $0 \leq x, y < 6, 1 \leq a < 6, 0 \leq b < 6$ und $h_{a,b}(x) = ((ax + b) \bmod 6) \bmod 4$. Kommentieren Sie das Ergebnis. Nehmen Sie insbesondere dazu Stellung, ob es sich auch jetzt um eine universelle Familie von Hashfunktionen handelt.

Aufgabe H14 (2+8+2 Punkte)

Am Ende dieser Aufgabe und auf der Webseite finden Sie ein Programm, das eine Datei lesen kann, welche je Zeile ein Wort enthalten soll. Das Programm zählt, wie viele verschiedene Wörter es in der Datei gibt, wobei es eine Hashtabelle verwendet.

- Wenn Sie das Programm auf einer typischen Datei laufen lassen, wie lang ist die Laufzeit auf Ihrem Computer typischerweise für eine Datei mit n Bytes?
- Erzeugen Sie eine Eingabedatei, deren Größe 1MB nicht überschreitet und deren Bearbeitung länger als 5 Sekunden dauert. Machen Sie die Datei Ihrem Tutor zugänglich. Erklären Sie, wie sie die Datei erstellt haben.
- Könnten Sie solch eine furchtbare Datei auch dann erstellen, wenn im Programm h nicht mit x sondern mit y multipliziert würde? (Eine sehr kurze Antwort reicht hier.)

```
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.util.*;

public class WordCount {
    static int x = 37;
    static int y = Math.abs((new Random()).nextInt());

    public static int hash(String w) {
        int h = 0;
        for(int i = 0; i < w.length(); i++) {
            h = h * x + w.charAt(i);
        }
        return Math.abs(h);
    }

    public static void main(String args[] throws IOException {
        String filename = args[0];
        List<String> words = Files.readAllLines(Paths.get(filename));
        int m = 100000;
        long count = 0;
        ArrayList<LinkedList<String>> slot = new ArrayList<>();
        for(int i = 0; i < m; i++) {
            slot.add(new LinkedList<String>());
        }
        for(String w : words) {
            LinkedList<String> list = slot.get(hash(w)%m);
            if(!list.contains(w)) {
                count = count + 1;
                list.add(w);
            }
        }
        System.out.println(count + " different words in " + filename + ".");
    }
}
```

Aufgabe H15 (3+3+3 Punkte)

Geben sie jeden Schritt im Detail an, wenn wir in folgender Skiplist nach 22, 40 und 0 suchen.

