

Übung zur Vorlesung Algorithmen und Datenstrukturen

Aufgabe T10

Betrachten Sie dieses Programm:

```
public class WasMachtDas {  
    public static void main(String[] args) {  
        int n = 1000;  
        boolean notp[] = new boolean[n];  
        for(int k = 2; k < n; k++) {  
            for(int i = 2 * k; i < n; i += k) notp[i] = true;  
        }  
        for(int i = 2; i < n; i++) {  
            if(!notp[i]) System.out.println(i);  
        }  
    }  
}
```

- Was macht dieses Programm?
- Wie groß ist die Laufzeit abhängig von n ?
- Wenn in der äußeren Schleife die Abbruchbedingung durch $i \leq \sqrt{n}$ ersetzt würde, würde sich die Laufzeit um mehr als einen konstanten Faktor verringern?

Hinweis: $\sum_{k=1}^n 1/k = H_n = \ln n + \gamma + O(1/n)$ mit $\gamma \approx 0.5772156649$.

Aufgabe T11

Entwerfen Sie eine gute Hashfunktion für binäre Suchbäume.

Aufgabe H8 (8 Punkte)

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 ein 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 H9 (16 Punkte)

Am Ende dieses Blatts und auf der Webseite finden Sie ein Programm, das eine Datei lesen kann, welche je Zeile ein Wort enthalten soll. Das Programm zählt, wieviele 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.)

```
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 + ".");
    }
}
```