

Übungen zur Vorlesung
Effiziente Algorithmen und Komplexitätstheorie
Sommersemester 2008
Blatt 14

Kurzaufgabe Was ist offenes und uniformes Hashing?

Aufgabe 14.1 (4 Punkte) Wir betrachten offenes und uniformes Hashing bei Tabellengröße m und analysieren die Datenstruktur bei $n \leq m$ gespeicherten Schlüsseln.

- Wie groß ist die erwartete Listenlänge pro Tabelleneintrag?
- Betrachte einen beliebigen Tabelleneintrag. Zeige: Die Wahrscheinlichkeit, dass die Listenlänge dieses Eintrags mindestens k beträgt, ist höchstens $1/k!$.
- Benutze Aufgabenteil b), um zu zeigen, dass die erwartete Länge der *längsten* Liste durch $O(\log m / \log \log m)$ nach oben beschränkt ist.

Hinweise: Die Ungleichungen $\binom{m}{k} \leq m^k/k!$ und $k! \geq (k/e)^k$ können hilfreich sein.

Kurzaufgabe Was bedeutet (c, k) -universell?

Aufgabe 14.2 (4 Punkte) Wir betrachten Cuckoo-Hashing. Gib zwei verschiedene, wie in der Vorlesung eingeschränkte Hashfunktionen f_1 und f_2 , eine Tabellengröße m sowie eine Folge von Schlüsseln an, sodass das Cuckoo-Hashing ohne Abbruchbedingung beim Einfügen des letzten Schlüssels in eine Endlosschleife gerät.

Hinweis: Wir können vereinfachend davon ausgehen, dass für Cuckoo-Hashing die $(2, k)$ universellen Funktionen der Gestalt

$$h(x) := \left(\sum_{i=0}^{k-1} a_i \cdot x^i \right) \bmod p$$

mit $a_i \in \{0, \dots, p-1\}$ für $i = 0, 1, \dots, k-1$ benutzt werden.

Aufgabe 14.3 (4 Punkte) Als ebenfalls in der Bioinformatik wichtiges Problem gilt *Longest Common Subsequence (LCS)*. Ein String (z_1, \dots, z_k) heißt Subsequenz eines Strings (x_1, \dots, x_n) , wenn eine streng monoton wachsende Folge i_1, \dots, i_k existiert, sodass $z_j = x_{i_j}$ für $j \in \{1, \dots, k\}$ gilt. Bei LCS besteht die Aufgabe darin, für zwei Strings $x = (x_1, \dots, x_n)$ sowie $y = (y_1, \dots, y_m)$ die maximale Länge k eines Strings zu finden, der sowohl Subsequenz von x als auch Subsequenz von y ist. Entwerfe mit dem Ansatz der dynamischen Programmierung einen effizienten Algorithmus, der das Problem LCS löst.

Prüfungsaufgabe

- Beschreibe Cuckoo-Hashing.
- Analysiere Cuckoo-Hashing.