Datenstrukturen und Algorithmen

Abgabe 7
Abgabe: 21.06.2017

Georg C. Dorndorf Matr.Nr. 366511
Adrian C. Hinrichs Matr.Nr. 367129

# 4	# 5	# 6	# 7	Σ

Aufgabe 4

Die Funktion f ist einfach zu berechnen. Sie ist außerdem surjektiv, da für alle $x \in [0,9]$ gilt: $\left\lfloor \frac{x \cdot 9 + y}{9} \right\rfloor = x \,\forall\, y \in [0,8]$. Der abgebildete Bereich ist also exakt $[0,89]$. Die Funktion verteilt die Schlüssel offensichtlich mit gleicher Häufigkeit auf den Zielbereich.

Die Funktion g ist einfach zu berechnen, allerdings nicht surjektiv. Das Bild von $A := \{1,2,4,8,16,32,64,...\}$ ist offensichtlich $g(A) = \{1,2,4,3,1,2,4,3,...\} = \{1,2,4,3\}$. Es ist zu erkennen, dass sich für $x \in \mathbb{N}$ die Funktionswerte von $g(x)$ zyklisch wiederholen und somit nicht den gesamten Funktionbereich treffen. Dementsprechend verteilt die Funktion g die Schlüssel nicht mit gleicher Häufigkeit auf den Zielbereich. Ähnliche (aber nicht identische) Schlüssel werden dabei relativ breit auf den Zielbereich verteilt.

Die Funktion h ist einfach zu berechnen und offensichtlicherweise surjektiv. Die Schlüssel werden von der Funktion nicht exakt, aber relativ, gleichmäßig auf den Zielbereich verteilt. Es ist lediglich für $x \in \{92,...,100\}$ $|h^{-1}(x)| = 9$ für $x \in \{0,...,91\}$ gilt jedoch $|h^{-1}(x)| = 10$. Ähnliche Schlüssel werden von dieser Funktion (bis auf die Ausnahme von Schlüsseln, die nahe bei einander sind und für die gilt $k_1 = 101 * \mathbb{N} - 1, k_2 = 101 * \mathbb{N}$) nicht breit gestreut.

Die Funktion i ist relativ einfach zu berechnen, surjektiv und verteilt die Schlüssel gleichmäßig auf den Zielbereich. Es gilt nämlich für $x \in \mathbb{N}$ $f : \mathbb{N} \to \mathbb{N}, x \mapsto \left\lfloor \frac{x}{2} \right\rfloor$, dass $f(\mathbb{N}) = \mathbb{N}$ ist.

Aufgabe 5

a)

$$\begin{aligned} h(k,i) &= (h_1(k) + i \cdot h_2(k)) \bmod 11 && \text{(I)} \\ &= (k \bmod 11 + i \cdot (k \bmod 9)) \bmod 11 && \text{(II)} \end{aligned}$$

b)

Notation Leere Zellen in den Hash–Tabellen enthalten den wert `NULL`, mit `d` beschriebene Zellen enthalten den String `DELETED`.

Sei die Hash–Tabelle definiert als T, das Objekt mit dem Schlüssel i (mit $i \in [0,10]$) ist dann $T[i]$.

Lösung

Abbildung 1: Initial leere Hash–Tabelle

Sondierungsschritte:

1. $h(97,0) = 9; T[9] =$ `NULL` $\Rightarrow$Einfügen

Abbildung 2: Hash–Tabelle nach einfügen von 97

									97	

Sondierungsschritte:

1. $h(15,0) = 4; T[4] =$ `NULL` $\Rightarrow$Einfügen

Abbildung 3: Hash–Tabelle nach einfügen von 15

				15					97	

Sondierungsschritte:

1. $h(42,0) = 9; T[9] = 97$
2. $h(42,1) = 4; T[4] = 15$
3. $h(42,2) = 10; T[10] =$`NULL` $\Rightarrow$ Einfügen

Abbildung 4: Hash–Tabelle nach einfügen von 42

				15					97	42

Sondierungsschritte:

1. $h(114,0) = 4; T[4] = 15$
2. $h(114,1) = 10; T[10] = 42$
3. $h(114,2) = 5; T[5] =$`NULL` $\Rightarrow$ Einfügen

Abbildung 5: Hash–Tabelle nach einfügen von 114

				15	114				97	42

Sondierungsschritte:

1. $h(15, 0) = 4; T[4] = 15 \Rightarrow$ Löschen

Abbildung 6: Hash–Tabelle nach Löschen von 15

				d	114				97	42

Sondierungsschritte:

1. $h(42, 0) = 9; T[9] = d \neq 114$
2. $h(42, 1) = 4; T[4] = d \neq 114$
3. $h(42, 2) = 10; T[10] = 42 = 42 \Rightarrow$Löschen

Abbildung 7: Hash–Tabelle nach Löschen von 42

				d	114				97	d

Sondierungsschritte:

1. $h(114, 0) = 4; T[4] = d \neq 114$
2. $h(114, 1) = 10; T[10] = d \neq 114$
3. $h(114, 2) = 5; T[5] = 114 = 114 \Rightarrow$Löschen

Abbildung 8: Hash–Tabelle nach Löschen von 114

				d	d				97	d

Aufgabe 6

a)

Die Hash-Funktion $h(k)$, die den Hash von k für ein Array x der Länge 7 berechnent, ist:

$$\text{Sei } hash := (\bar{k} \bmod 7) \cdot 11 + 5) \bmod 7 \tag{III}$$

$$h(k) = \begin{cases} hash & \text{, falls x[hash] leer} \\ h(hash) & \text{, sonst} \end{cases} \tag{IV}$$

b)

Die in a) beschriebene Hash-Funktion ist vergleichsweise schwer zu berechnen beziehungsweise besteht aus mehreren für die Streuung irrelevanten Teilen. Des weiteren ist es möglich, dass trotz des rekursiven rehashens bei Kollisionen keine 7 unterschiedlichen Elemente in der Hashtabelle gespeichert werden können und die Hash-Funktion nicht terminiert, da $h(3) = 3$ gilt.

Asymptotisch betrachtet ist die Quersumme von $x \in \mathbb{N}$ in $\mathbb{N}$ gleichverteilt. Um die Streuung zu analysieren genügt es folglich, den Teil der Hashfunktion h zu betrachten, der von der Quersumme unabhängig ist. Sei dazu $h'(k') := (k' \cdot 11 + 5) \bmod 7$ mit $k' \in \{0, ..., 6\}$. Die hierraus resultierende Abbildung ist eine Bijektion und es ist: $h'(0) = 5$, $h'(1) = 2$, $h'(3) = 3$, $h'(4) = 0$ $h'(5) = 4$, $h'(6) = 1$. Es ist zu erkennen, dass, da die Quersumme in $\mathbb{N}$ gleichverteilt [1] also auch die Quersumme modulo 7 gleichverteilt ist, die Hash-Funktion aus a) relativ gut streut.

Die Funktion ist trotzdem nicht als Hash-Funktion geeignet, da sie über kein vernünftiges Kollisionshandling verfügt und möglicherweise nicht terminiert.

Aufgabe 7

a)

Behauptung:: Behauptung:: Falls wir in K_1 stecken bleiben, so haben wir jeden an K_1 angrenzenden Gang in beiden Richtungen durchlaufen.

Beweis per Wiederspruch.: Beweis per Wiederspruch.:

Gegeben: Gegeben: Wir sind in K_1 stecken geblieben.

Anngenommen, dass: Anngenommen, dass: dass wir nicht jeden Gang der an K_1 grenzt in jede Richtung einmal betreten hätten.
Dann gilt (mindestens) einer der beiden folgenden Fälle:

1. (Mindestens) ein Gang ist in die Richtung von K_1 weg noch nicht durchlaufen worden. In diesem Fall können wir K_1 über den Gang verlassen, und stecken somit nicht fest. ϟ

2. (Mindestens) ein Gang ist in die Richtung zu K_1 hin noch nicht betreten worden. Sei m die Anzahl der Gänge zu K_1, welche in die Hin–Richtung bereits betreten worden sind, l die Anzahl der Gänge über die K_1 verlassen worden ist und n die Gesamt–Anzahl der Gänge zu K_1. Nun gilt $m < n$. Nach Fall 1 ist n jedoch auch die Anzahl der Gänge, über welche K_1 bereits verlassen worden ist ($n = l$). Da wir auf K_1 starten und keinen Gang doppelt betreten dürfen, muss aber die Anzahl der Gänge über die K_1 verlassen worden ist gleich m oder gleich $m+1$ sein, also muss gelten $l \leq m$. Insgesamt ergibt sich also $m < l \leq m \Leftrightarrow m < m$ ϟ

Die Annahme ist also falsch, die Behauptung muss gelten.

QED

b)

Behauptung:: Behauptung:: Falls wir in K_1 stecken bleiben, so haben wir jeden Gang, der an eine von uns besuchte Kreuzung angrenzt, in beiden Richtungen durchlaufen.

Beweis per Wiederspruch.: Beweis per Wiederspruch.:

[1] https://de.wikipedia.org/wiki/Quersumme

Georg C. Dorndorf Matr.Nr. 366511
Adrian C. Hinrichs Matr.Nr. 367129

Gegeben: **Gegeben:** Wir sind auf K_1 stecken geblieben.

Anngenommen, dass: **Anngenommen, dass:** wir nicht alle Gänge, die an eine bereits besuchte Kreuzung grenzt in beide Richtungen durchlaufen hätten.
Nach Aufgabe 7.a) und Globalübung 3.b) gilt, dass jeder Gang zu K_1 in jede Richtung genau ein mal durchlaufen worden ist. Es muss also eine Kreuzung $K_i \neq K_1$ existieren, welche besucht worden ist, aber mindestens ein an sie angrenzender Gang noch nicht in beide Richtungen durchlaufen worden ist. Nach Regel R3 und Globalübung 3.b) darf nun keiner der Gänge auf dem gegangenen Weg zwischen K_i und K_1 bereits durchlaufen worden sein (Von K_1 aus betreten wir jeden Knoten zum ersten mal). Dies bedeutet, einer der Gänge, welcher an K_1 grenzt wäre zum Zeitpunkt des Steckenbleibens nur einmal durchlaufen worden. Dies steht im Widerspruch zu Aufgabe 7.a).ϟ

Die Annahme ist also falsch, die Behauptung muss gelten.

QED

c)

Behauptung:: **Behauptung::** Wir entdecken irgendwann einmal den Ausgang.

Beweis per Wiederspruch.: **Beweis per Wiederspruch.:**

Anngenommen, dass: **Anngenommen, dass:** wir den Ausgang nicht Gefunden hätten.
Nach Globalübung 3.c) bleiben wir dann auf K_1 stecken. Nach Aufgabe 7.b) haben wir dann jeden an einen Besuchten knoten angrenzenden Gang von beiden Seiten betreten, das heißt wir hätten beide an diesen Gang angrenzende Knoten betreten. Per struktureller Induktion folgt, dass wir alle Knoten des Labyrinthes betreten haben. Da der Ausgang (bzw. das Ziel) aber auch ein Knoten ist, hätten wir das Ziel betreten (also finden) müssen. ϟ

Die Annahme ist widerlegt, unsere Behauptung muss also gelten.

QED