

Algorithmen und Datenstrukturen

Sommersemester 2022

Blatt 7

Kevin Angele, Tobias Dick, Oskar Neuhuber,
Andrea Portscher, Monika Steidl, Laurin Wischounig

Abgabe bis 10.05.2022 23:59
Besprechung im PS am 12.05.2022

Aufgabe 1 (1 Punkt): Einfügen in Hashtabellen

Beim Einfügen in eine Hashtabelle wird der Hashcode des einzufügenden Elements verwendet, um festzulegen, an welcher Stelle im zugrundeliegenden Array es gespeichert werden soll. Gehen Sie von der Hashfunktion $h(x) = ((x \ll 2) \text{ xor } (x + 13))$ aus und fügen sie folgende Werte in eine anfangs leere Hashtabelle mit Größe 10 ein:

3	2	62	0	11	9
---	---	----	---	----	---

. Verwenden Sie als Kompressionsfunktion $c(x) = x \bmod 10$

Lösung:

3 -> 28 -> 8

2 -> 7

62 -> 179 -> 9

0 -> 13 -> 3

11 -> 52 -> 2

9 -> 50 -> 0

9		11	0			2	3	62	
---	--	----	---	--	--	---	---	----	--

Aufgabe 2 (2 Punkte): Hashfunktionen

Beschreiben Sie, wieso die beiden folgenden Hashfunktionen $a(x)$ und $b(x)$ für eine Hash-Tabelle der Größe N , in der ganzzahlige Schlüssel gespeichert werden, zu Problemen führen können. N ist hierbei eine Primzahl.

1. $a(x) = \lfloor \frac{x}{N} \rfloor \bmod N$

2. $a(x) = \bar{x}$ (\bar{x} bezeichnet hierbei die Quersumme)

Lösung:

1. Benachbarte Zahlen werden hier durch das Runden von $\frac{x}{N}$ extrem häufig auf die gleichen Hashwerte abgebildet, was zum Auftreten einer großen Anzahl an Kollisionen führt.

2. Für beliebig große ganzzahlige Schlüssel wächst auch deren Quersumme über eine beliebige Grenze, in unserem Fall über N , hinaus. Alle Hashwerte über dieser Grenze können nicht mehr zur Adressierung in die Hashtabelle eingefügt werden. Es fehlt eine Kompressionsfunktion. Zudem liegen die Hashwerte benachbarter Zahlen direkt nebeneinander, wodurch eine geringe Streuung entsteht.

Aufgabe 3 (2 Punkte): Invertierter Index

Ein invertierter Index ist ein Mapping von Inhalten in einem Dokument zu ihrem Ort im Dokument. Schreiben sie eine Funktion, die für einen String einen solchen invertierten Index erstellt. Der Index soll es ermöglichen, für ein beliebiges Wort herauszufinden, ob es im String vorkommt und falls ja, in welcher Zeile bzw. in welchen Zeilen. Verwenden Sie den Code in `InvertedIndex.java` als Vorlage. Sie können/sollten eine Map aus der Javastandardbibliothek verwenden.

Lösung:

Siehe: `InvertedIndexSolution.java`

Aufgabe 4 (5 Punkte): Anzahl verschiedener Buchstaben

Schreiben sie eine Funktion, die alle in einem String vorkommenden Wörter nach der Anzahl der verschiedenen im Wort vorkommenden Buchstaben sortiert und ausgibt. Die Funktion soll zusätzlich zu jedem Wort die Anzahl der verschiedenen Buchstaben ausgeben. Verwenden Sie den Code in `UniqueCharacterCount.java` als Vorlage. Sie können/sollten eine Map aus der Javastandardbibliothek verwenden.

Beispiele:

“ab” > “aa”

“string” > “text”

“java” = “text”

Lösung:

Siehe: `UniqueCharacterCountSolution.java`