Informatik II: Algorithmen und Datenstrukturen SS 2017

Vorlesung 5a, Dienstag, 23. Mai 2017
(Universelles Hashing, Teil 1)

Prof. Dr. Hannah Bast
Lehrstuhl für Algorithmen und Datenstrukturen
Institut für Informatik
Universität Freiburg

Blick über die Vorlesung heute

- Organisatorisches
  - Feedback Tutoren zum ÜB3 — O-Notation
  - Erfahrungen mit dem ÜB4 — eigene Hash Map
  - Mathe — Kurzanleitung
- Inhalt
  - Universelles Hashing — Erklärung + Theorem
  - Wahrscheinlichkeitsrechnung — Crash-Kurs
  - ÜB5: Berechnen sie die Kollisionswahrscheinlichkeiten für 4 Klassen von Hashfunktionen (aus der Vorlesung 5b)

  Achtung: es gibt in der Klausur **mit Sicherheit** eine Aufgabe zu universellem Hashing (wie bisher auch immer)

Feedback Tutoren zum ÜB3

- Rückmeldung aus den Korrekturen

  $g: n \mapsto n$

  - Öfter sowas wie $O(n^2 + n + 10)$ anstatt $O(n^2)$
  - Das n_0 wurde oft vergessen bzw. nicht explizit genannt
  - Probleme mit Implikationspfeilen und ihrer Richtung
  - Redundante Semikolons im Python-Code auf dem ÜB3
  - Bei den Aufgaben 2 oder 3 oft wenig oder keine Begründung

    Sowas gibt in der Klausur definitiv (erheblichen) Punktabzug
  - Man schreibt $n = O(n) = O(n^2)$ statt $f \in O(n) \subseteq O(n^2)$
  - Die Ableitung von 2^n nach n ist definitiv nicht $n \cdot 2^{n-1}$
  - Immer hinschreiben: 1. Was ist gegeben, 2. was ist zu zeigen

Erfahrungen mit dem ÜB4

- Zusammenfassung / Auszüge
  - Die Aufgabe (und die Anwendung) hat vielen Spaß gemacht
  - Endlich wieder ein ÜB mit Programmieren
  - Dankbar für den Code zum Auslesen der Wörter
  - Probleme mit Gnuplot … habe ich aber in VL 1a vorgemacht
  - "Aufgabe 1: ging sau schnell, war nur falsch"
  - "Fand es schwierig zu verstehen, was man von mir wollte"
  - "Live-Coding relativ langweilig im Vergleich zur Theorie, aber beim Rest scheint es ja gut anzukommen"
  - Rückmeldung von Tutor/in sehr hilfreich

Mathe

eine Ungleichung $\underline{A} \leq B$

eine Gleichung $\underline{A} = B$

eine Implikation $\underline{A} \Rightarrow B$

linke Seite

- Kurzanleitung
  - Hinschreiben: 1. Was ist gegeben? + 2. Was ist zu zeigen?
  - Variante 1: "normaler" Beweis

    Die linke Seite von "was ist zu zeigen" hinschreiben + darauf anwenden, was sonst noch gegeben ist + Ausdrücke vereinfachen … bis man zu dem kommt, was gezeigt werden soll
  - Variante 2: Widerspruchsbeweis viel seltener

    Annehmen, dass das Gegenteil von dem gilt, was zu zeigen ist + umformen wie oben … bis man zu etwas kommt, was nicht sein kann (aber nicht weil man sich verrechnet hat!)
  - Die meisten Beweise in dieser Vorlesung brauchen eine oder keine Idee, der Rest geht im Wesentlichen "automatisch"

Universelles Hashing 1/10

im ÜB4: die Wörter

- Zur Erinnerung

  oder mehr oder weniger zufällig

  - Wenn die Schlüsselmenge zufällig ist, tut es auch die einfache Hashfunktion $h(x) = x \bmod m$
  - Für bestimmte Schlüsselmengen kann diese Funktion dagegen beliebig schlecht sein, zum Beispiel

    $h(x) = x \bmod 10$ und $x = 12, 42, 32, 72, 102, \ldots$

    h(x) 2 2 2 2 2

  - Allgemeiner: keine einzelne Hashfunktion h kann für alle Schlüsselmengen gut sein, weil:

    z.B. alle Wörter

    h ist Funktion von U nach $\{0, \ldots, m-1\}$ und $|U| \gg m$

    Selbst im besten Fall werden so $|U| / m$ Schlüssel auf denselben Wert abgebildet

Universelles Hashing 2/10

z.B. $m = 10$, $|U| = 100$, $p = 101$

dann wäre z.B.

$g(x) = (57x + 83) \bmod 101 \bmod 10$

eine Fkt. aus der Klasse unten

p ist eine Primzahl mit $p \geq |U|$

- **Idee**
  - Eine Menge (Klasse) von Hashfunktionen zur Auswahl
  - Und zwar so, dass man leicht ein zufälliges Element aus dieser Menge wählen kann
  - Beispiel: $h(x) = a \cdot x \bmod m$ mit $a \in \{0, \ldots, m-1\}$

    Das sind (nur) m verschiedene Hash-Funktionen

    Wir sehen morgen, dass das keine gute Klasse ist

    $U = \{0, \ldots, |U|-1\}$

    fix für die Klasse
  - Beispiel: $h(x) = (a \cdot x + b) \bmod p \bmod m$ mit $a, b \in U$

    Das sind $|U|^2$ verschiedene Hash-Funktionen

    Wir sehen morgen, dass das eine sehr gute Klasse ist

    Wie findet man $p \geq |U|$, p prim?

    ANTWORT: gute Frage! siehe ÜB5 vom letzten Jahr.

Universelles Hashing 3/10

nicht: eine einzelne Hash-Funktion

- Was ist eine gute Klasse (informal)
  - Eine Klasse H von Hashfunktionen ist dann gut, wenn:

    Für jede Schlüsselmenge S gibt es viele Funktionen in H, die S "möglichst gut über die Hashtabelle verteilen"

    Das machen wir auf den nächsten Folien formaler
  - Dann können wir einfach eine zufällige Funktion aus H wählen und hoffen, dass alles gut klappt

    Wenn nicht, merken wir das und machen nach einer Weile einen Rehash, mit einer anderen Funktion aus H

    Wenn H die obige Eigenschaft hat, wird das nicht oft passieren und "im Mittel" gut funktionieren

Universelles Hashing 4/10

z.B. $m = 10$, $|S| = 50$

$\Rightarrow |S|/m = 5$

- Zufälliges Werfen
  - Schlüsselmenge S, Hashtabelle T mit m "Plätzen"
  - Die beste Art S möglichst gut über T zu verteilen" ist, wenn jeder Platz genau |S| / m Schlüssel abbekommt
  - Das erreicht man mit **zufälligem Werfen**:

    Für jedes x ∈ S, wähle einen zufälligen Platz in T

    Dann ist die erwartete Anzahl Elemente an einem bestimmten Platz genau |S| / m
  - Das beweisen wir jetzt erstmal

    Vorher ein Crash-Kurs Wahrsch.keitsrechnung (Folien 15–18)

    Zur Auffrischung oder um es zum ersten Mal zu verstehen

Universelles Hashing 5/10

- Zufälliges Werfen, Verteilung
  - Schlüsselmenge S, Hashtabelle mit m Plätzen
  - Sei $h(x)$ der Platz von Schlüssel x
  - Sei $S_i = \{ x \in S : h(x) = i \}$
  - Wir zeigen jetzt, dass $E(|S_i|) = |S| / m$

Wir schauen uns S_i für ein festes i an.

Definiere Indikatorvariable $I_x = 1$ genau dann wenn $h(x) = i$

$I_x = 0$ sonst.

$$E(I_x) \overset{\text{Folie 19}}{=} Pr(I_x = 1) = \frac{1}{m}$$

$|S_i| = \sum_{x \in S} I_x$ ... wie auf Folie 19

$$\Rightarrow \quad E(|S_i|) = E\Big(\sum_{x \in S} I_x\Big) = \sum_{x \in S} \underbrace{E(I_x)}_{= 1/m} = |S|/m$$

Universelles Hashing 6/10

- Zufälliges Werfen, Problem
  - "Zufälliges Werfen" ist keine gute Klasse von Hashfunktionen
  - Die Hashfunktionen, die wir uns bisher angeschaut haben, waren sehr leicht zu speichern und auszuwerten
  - Zum Beispiel: $h(x) = x \bmod m$

    Kann man in O(1) Platz speichern und in O(1) Zeit auswerten für einen beliebigen Schlüssel x aus dem Universum
  - Für einen zufälligen Wurf nicht so einfach:

    Man müsste sich für jeden Schlüssel im Universum (oder zumindest in S) merken, wo er hingeworfen wurde

    Denken Sie darüber nach: man bräuchte eine Hash Map, um das effizient abzuspeichern / darauf zuzugreifen

Universelles Hashing 7/10

- Ziel
  - Unser Ziel ist eine Klasse von Hashfunktionen für die gilt:

    1. Eine zufällige Funktion aus dieser Klasse verhält sich (fast) genauso gut wie zufälliges Werfen

    2. Man kann leicht eine zufällige Funktion auswählen, abspeichern und für einen Schlüssel x auswerten
  - Die Definition von universellem Hashing versucht genau Eigenschaft 1 formal zu fassen … siehe nächste Folie

Universelles Hashing 8/10

- Definition
  - Sei H eine Menge von Hashfunktionen $U \rightarrow \{0, ..., m-1\}$
  - H ist c-universell wenn für alle $x, y \in U$ mit $x \neq y$ gilt:

    $$| \{h \in H : h(x) = h(y)\} | \leq c \cdot |H| / m$$

    Heißt: x und y kollidieren nur unter einem Bruchteil von H
  - Mit anderen Worten, wenn $h \in H$ zufällig gewählt, dann

    $$\text{Prob}(h(x) = h(y)) \leq c \cdot 1 / m$$

  - Bei zufälligen Werfen gilt das mit $c = 1$ … und man kann zeigen, dass besser als $c = 1$ auch nicht geht

    Morgen sehen wir Klassen von Hash-Funktionen mit c = 1 oder c = 2, was für praktische Zwecke oft genauso gut ist

Universelles Hashing 9/10

- Zentraler Satz
  - Sei H eine c-universelle Klasse von Hashfunktionen
  - Sei S eine Menge von Schlüsseln und $h \in H$ zufällig gewählt
  - Für ein $x \in S$ sei S_x die Menge der Schlüssel y mit $h(y) = h(x)$
  - Dann ist $E(|S_x|) \leq 1 + c \cdot |S| / m$ ... $|S|/m$ wäre optimal

    $m = 100$, $|S| = 200$, $c = 2$ $\Rightarrow$ $E(|S_x|) \leq 1 + 2 \cdot \frac{200}{100} = 5$
  - Insbesondere: Falls $m = \Omega(|S|)$ gilt $E(|S_x|) = O(1)$
  - Das beweisen wir auf der nächsten Folie
  - Man beachte: die vermeintlich einfachere Aussage, dass $E(|S_i|) \leq 1 + c \cdot |S| / m$ gilt im Allgemeinen **nicht**

    Das macht aber nichts, für z.B. die Laufzeit von insert bei Hashing mit Verkettung reicht auch die Aussage von oben

Universelles Hashing 10/10

gegeben: h zufällig, $x \neq y$
$\Rightarrow Pr(h(x) = h(y)) \leq \frac{c}{m}$

- Beweis von $E(|S_x|) \leq 1 + c \cdot |S| / m$

zu zeigen $\underline{E(|S_x|)} \leq \ldots$

↳ damit fangen wir an (im Kopf)

$I_y = 1$ gdw $h(y) = h(x)$
$I_y = 0$ sonst
für $y \in S$

nach Def. universell

$E(I_y) = Pr(I_y = 1) \leq c/m$ ... für $y \in S \setminus \{x\}$
für $y = x$: $Pr(I_y = 1) = 1$

x persönlich

$$E(|S_x|) = E(1 + \sum_{y \in S \setminus \{x\}} I_y) = 1 + \sum_{y \in S \setminus \{x\}} \underbrace{E(I_y)}_{\leq c/m}$$

$$\leq 1 + c \cdot \frac{|S| - 1}{m}$$

$$\leq 1 + c \cdot |S| / m$$

Wahrscheinlichkeitsrechnung 1/4

WURF 1 WURF 2

$$\Omega = \{ (1,1), (1,2), \dots, (1,6), \\ (2,1), (2,2), \dots, (2,6), \\ (3,1), (3,2), \dots, (3,6), \\ (4,1), (4,2), \dots, (4,6), \\ (5,1), (5,2), \dots, (5,6), \\ (6,1), (6,2), \dots, (6,6) \}$$

- **Wahrscheinlichkeitsraum / Ereignisse**
  - Wir beschränken uns hier auf den diskreten Fall
  - Wahrscheinlichkeitsraum Ω von sog. Elementarereignissen
  - Die haben Wahrscheinlichkeiten ... Bedingung $\Sigma_{e \in \Omega} \Pr(e) = 1$
  - Ereignis E = Teilmenge von Ω, Wahrsch. $\Pr(E) = \Sigma_{e \in E} \Pr(e)$
  - Zum Beispiel: zweimal würfeln, dann $\Omega = \{1,\dots,6\}^2$

    Jedes e aus Ω hat dann Wahrscheinlichkeit $\Pr(e) = 1/36$

    E = beide Augenzahlen sind gerade, dann $\Pr(E) = \frac{3 \cdot 3}{36} = \frac{1}{4}$

    3 · 3 der Elementarereignisse oben haben diese Eigenschaft

Wahrscheinlichkeitsrechnung 2/4

für das X hier:

$$E(X) = \sum_{x=2}^{12} x \cdot Pr(X = x)$$

- Zufallsvariable
  - ... weist einem Ausgang des Zufallsexperiments eine Zahl zu
  - Zum Beispiel: X = Summe Augenzahlen bei zweimal Würfeln
  - Sowas wie X = 12 oder X ≥ 7 sind dann einfach Ereignisse
  - Beispiel 1: Prob(X = 2) = $\frac{1}{36}$ El. ereignisse: (1,1)
  - Beispiel 2: Prob(X = 4) = $\frac{3}{36} = \frac{1}{12}$ El. ereignisse: (1,3), (2,2), (3,1)
  - **Erwartungswert** ist definiert als **E**(X) = Σ k · Pr(X = k)

    Intuitiv: gewichtetes Mittel der möglichen Werte von X, wobei die Gewichte die Wahrscheinlichkeiten der entspr. Werte sind
  - Beispiel von oben: X = Summe Augenzahl zweimal Würfeln

$$E(X) = 2 \cdot \underbrace{Pr(X=2)}_{=\frac{1}{36}} + 3 \cdot \underbrace{Pr(X=3)}_{=\frac{1}{18}} + \ldots + 12 \cdot \underbrace{Pr(X=12)}_{=\frac{1}{36}}$$

Wahrscheinlichkeitsrechnung 3/4

- Summe von Erwartungswerten
  - Für beliebige (diskrete) Zufallsvariablen $X_1, \ldots, X_n$ gilt

    $\mathbf{E}(X_1 + \ldots + X_n) = \mathbf{E}(X_1) + \ldots + \mathbf{E}(X_n)$

    Gilt auch, wenn die $X_1, \ldots, X_n$ **nicht** unabhängig sind!
  - Beispiel: Summe Augenzahl bei zweimal Würfeln
  - Sei X_1 = Augenzahl Würfel 1 und X_2 = Augenzahl Würfel 2
  - Sei $X = X_1 + X_2$ die Summe der Augenzahlen

$\left(\sum_{i=1}^{n} i = \frac{1}{2} n(n+1)\right)$

$E(X_1) = 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + 3 \cdot \frac{1}{6} + \cdots + 6 \cdot \frac{1}{6} = \frac{1+2+\cdots+6}{6} = \frac{6 \cdot 7/2}{6} = 3.5$

$E(X_2) = 3.5$

(der wunderbare Satz von oben

$E(X) = E(X_1 + X_2) = E(X_1) + E(X_2) = 3.5 + 3.5 = 7$ $\boxtimes$

Wahrscheinlichkeitsrechnung 4/4

- Summe von Erwartungswerten, Korollar
  - Bei einem Zufallsexperiment tritt das Ereignis E mit Wahrscheinlichkeit p auf. Sei X die Anzahl der Auftreten von E bei n Ausführungen dieses Experimentes, dann ist **E**(X) = n · p
  - Beispiel: **E**(Anzahl Sechser bei 60 mal Würfeln) = 10
  - Beweis: Definiere eine sogenannte **Indikatorvariable**

    $I_j = 1$ wenn E eintritt bei Ausführung j, sonst $I_j = 0$

$Pr(I_j = 1) = p$ ... im Beispiel oben: $\frac{1}{6}$ = eine 6 gewürfelt

$Pr(I_j = 0) = 1-p$ ... im Beispiel oben: $\frac{5}{6}$ = keine 6 gew.

↳ nach Def. E-Wert

$E(I_j) = 1 \cdot \underbrace{Pr(I_j = 1)}_{=p} + 0 \cdot \underbrace{Pr(I_j = 0)}_{=1-p} = p \quad (= Pr(I_j = 1))$

$X = \sum_{j=1}^{n} I_j$ ... genau deswegen hat man I_j so definiert

↳ Satz von Folie vorher

$E(X) = E\left(\sum_{j=1}^{n} I_j\right) = \sum_{j=1}^{n} E(I_j) = \sum_{j=1}^{n} p = n \cdot p$ ■

Literatur / Links

- Universelles Hashing
  - In Mehlhorn / Sanders:

    4 Hash Tables and Associative Arrays
  - In Wikipedia

    http://en.wikipedia.org/wiki/Universal_hashing