

Algorithmen 1 SS 2013 – Tutorium 7

6. Tutorium

Sarah Lutteropp

28. Mai 2013

Übersicht

- 1 Übungsblätter
- 2 Kreativaufgabe
- 3 Probe-Probeklausur

4. Übungsblatt

Abschreiben ...

Gibt immernoch 0 Punkte.

Aufgabe 1.a

Bei Hashing mit verketteten Listen neue Elemente an den Anfang der Liste einfügen.

Aufgabe 1.c

Erwartete Suchkosten ! = Suchkosten im schlechtesten Fall:

Erwartete Kosten = (Anzahl der Operationen, die man gemacht hat)/(Anzahl der Elemente)

4. Übungsblatt

Aufgabe 2

Noch sowas von wegen zufällige Hashfkt. aus universeller Familie hinschreiben.

Aufgabe 3

Hashtabelle benutzen. Was, wenn die Schlüssel irgendwelche Objekte sind oder wild durcheinandergeschmissene Zahlen?

4. Übungsblatt

Möchte jemand vorrechnen?

Kreativaufgabe

Gegeben sei ein Universum von Elementen die in einer Prioritätsliste verwaltet werden sollen. Die Wichtigkeit eines Elements hängt nur von seinem Schlüssel ab, wobei es nur k verschiedene Schlüssel geben soll. Man kennt den Algorithmus, der die Prioritätsliste verwenden soll, so gut, dass man weiß, dass alle Einfügungen eine geringere Priorität als das zuletzt entnommene Element haben. Finden Sie eine Datenstruktur, in die neue Elemente in $\mathcal{O}(1)$ eingefügt werden können, und die Entnahme amortisiert in $\mathcal{O}(\frac{k}{n} + 1)$ funktioniert. n sei dabei die Gesamtzahl jemals eingefügter Elemente. Interessant insbesondere für $n > k$.

Kreativaufgabe

Gegeben sei ein Universum von Elementen die in einer Prioritätsliste verwaltet werden sollen. Die Wichtigkeit eines Elements hängt nur von seinem Schlüssel ab, wobei es nur k verschiedene Schlüssel geben soll. Man kennt den Algorithmus, der die Prioritätsliste verwenden soll, so gut, dass man weiß, dass alle Einfügungen eine geringere Priorität als das zuletzt entnommene Element haben. Finden Sie eine Datenstruktur, in die neue Elemente in $\mathcal{O}(1)$ eingefügt werden können, und die Entnahme amortisiert in $\mathcal{O}(\frac{k}{n} + 1)$ funktioniert. n sei dabei die Gesamtzahl jemals eingefügter Elemente. Interessant insbesondere für $n > k$.

Kreativaufgabe

Gegeben sei ein Universum von Elementen die in einer Prioritätsliste verwaltet werden sollen. Die Wichtigkeit eines Elements hängt nur von seinem Schlüssel ab, wobei es nur k verschiedene Schlüssel geben soll. **Man kennt den Algorithmus, der die Prioritätsliste verwenden soll, so gut, dass man weiß, dass alle Einfügungen eine geringere Priorität als das zuletzt entnommene Element haben.** Finden Sie eine Datenstruktur, in die neue Elemente in $\mathcal{O}(1)$ eingefügt werden können, und die Entnahme amortisiert in $\mathcal{O}(\frac{k}{n} + 1)$ funktioniert. n sei dabei die Gesamtzahl jemals eingefügter Elemente. Interessant insbesondere für $n > k$.

Probe-Probeklausur

Wird ausgeteilt.

Bis zum nächsten Mal.

