Universität Stuttgart
Fakultät 5, Institut FMI
Einführung in die Informatik III

Wintersemester 2007/2008
Claus, Riexinger
06.01.2008

Aufgabenblatt 10 **Abgabe: 13.01.2008, 24.00 Uhr**

---

1. Hashing - alte Prüfung (leicht) (4 Punkte)

Wir betrachten offenes Hashing mit linearer und quadratischer Kollisionsstrategie. Es soll eine Folge von 7 Elementen in ein Feld mit Platz für 9 Elemente eingetragen werden.

Es soll nun die Folge REDUNDANT, TEXTERKENNUNG, INVARIANTE, CLAUS, INITIALISIERUNG, HAPLOID, GRAPH mit der Hashfunktion

$$f(a_1 a_2 .. a_r) = (\phi(a_1) + \phi(a_3) + 6) \bmod 9$$

(wobei $\phi(x)$ wie üblich die Position des Buchstabens x im Alphabet angibt.) in die Hashtabelle eingetragem werden.

a) Tragen Sie die Folge in eine Hashtabelle (A) ein, verwenden Sie dazu eine lineare Kollisionsstrategie. (Die Verschiebung c sei hierbei 1.) (2 P)

b) Tragen Sie die Folge in eine Hashtabelle (B) ein, verwenden Sie dazu eine quadratische Kollisionsstrategie. (2 P)

2. Hashing I (leicht) (2 + 2 + 2 Punkte)

In eine Hashtabelle mit 11 Einträgen (Indizes 0 bis 10) sollen die Wörter "CLAUS", "DIASHOW", "LILA", "SUMME", "GELB', "GELEHRTER", "FRISCH", "POOL" und "BANK" in der angegebenen Reihenfolge eingefügt werden.
Für die verwendete Hashfunktion sind den Buchstaben A bis Z die Werte b(A)=1 bis b(Z)=26 zugeordnet; gewichten Sie den ersten Buchstaben mit 1 und den vierten Buchstaben mit Gewicht 3. Die Hashfunktion sei also $f(a_1 a_2 a_3 ...) = ((b(a_1) + 3 \cdot b(a_4)) \bmod 11)$ (Beispiel: der Hashwert des Wortes "FUSSBALL" ist damit $((6 + 3 \cdot 19) \bmod 11) = 8)$. Kollisionen sollen mit dem Verfahren der linearen Sondierung aufgelöst werden. Die Schrittweite ist 1.

a) Geben Sie die ausgefüllte Tabelle, sowie für jeden Eintrag alle die im Zuge der Kollisionsauflösung berechneten Indizes an.

b) Perfektes Hashing: modifizieren Sie die Gewichte in der Hashfunktion so, dass für die obigen 9 Elemente in einer Hashtabelle mit 11 Einträgen keine Kollision stattfindet.

c) Finden Sie auch so eine perfekte Hashfunktion, wenn die Hashtabelle nur 9 Einträge hat?

3. Hashing I (leicht) (2 Punkte)

Die Teilaufgabe c) der Aufgabe 2 ist nur mit viel Glück durch manuelles Probieren zu lösen. Daher wollen Sie ein kleines Programm schreiben, der alle Möglichkeiten durchprobiert. Aber wieviele Möglichkeiten für die Hashfunktion (Divisionsverfahren) gibt es, wenn Sie maximal die ersten vier Buchstaben des Wortes verwenden dürfen? Geben Sie im eClaus-System nicht nur die Lösung, sondern auch eine Begründung der Lösung an!

4. Hashing-Programm I (leicht) (4 Punkte)

Schreiben Sie ein Java-Programm, das eine Zeichenkette vom Benutzer einliest und diese Zeichenkette in einer Hashtabelle speichert (jede Eingabe soll dabei in Großbuchstaben umgewandelt werden). Verwenden Sie das geschlossene Hashing mit linearen Listen. Dabei soll die Hashtabelle nur aus Zeigern auf Elemente bestehen (zu Anfang alles Nullzeiger). Überlegen Sie sich selbst eine Hashfunktion und begründen Sie Ihre Entscheidung.

5. Hashing-Programm II (leicht) (4 Punkte)

Erweitern Sie Ihr Programm aus Aufgabe 4 mit der Option, Elemente aus der Hashtabelle zu löschen. Dabei soll das Programm dem Benutzer mitteilen, ob die zu löschende Zeichenkette in der Hashtabelle enthalten war oder nicht. Wieso ist das Löschen einfacher zu implementieren, wenn die Hashtabelle nur aus Zeigern besteht (und nicht aus Elementen mit angehängter Liste)?