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

Sommersemester 2006
Claus, Draskoczy, Schmid
23.06.06

Aufgabenblatt 9 **Abgabe: 01.07.06, 24.00 Uhr**

---

1. Quadratische Sondierung (leicht) (schriftlich, 2 + 2 Punkte)
In eine Hashtabelle mit 11 Einträgen (Indizes 0 bis 10) sollen die Wörter „CLAUS“, „DRASKOCZY“, „LEWANDOWSKI“, „SCHMID“, „GRUBER“, „GRIEPENTROG“, „FISCHER“„PHILIPP“ und „VOETTER“ (auf vielfachen Wunsch) 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 dritten Buchstaben mit Gewicht 2. Die Hashfunktion sei also $f(a_1a_2a_3...) = ((b(a_1)+2\cdot b(a_3)) \mod 11)$ (Beispiel: der Hashwert des Wortes „FUSSBALL“ ist damit $((6+2\cdot 19) \mod 11) = 0$). Kollisionen sollen mit dem Verfahren der quadratischen Sondierung (siehe Skript, Seite 52) aufgelöst werden.

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

b) Geben Sie bei jeder Kollision zusätzlich an, ob es sich dabei um eine Primär- oder Sekundärkollision handelt.

2. Rehashing (mittel) (schriftlich, 5 Punkte)
Im unten stehenden Beispiel ist die Hashtabelle ($p = 5$) zu 80% voll, daher soll ein Rehashing durchgeführt werden. Bei diesem Rehashing soll die Tabelle auf $p' = 11$ erweitert werden (Kollisionsbehandlung: lin. Sondieren).

Die Hashwerte der Funktion f bezüglich der ursprünglichen Tabelle lauten:
$f(A) = 4$, $f(B) = 0$, $f(C) = 2$, $f(D) = 3$ und $f(E) = 4$

Die Hashwerte der Funktion f_{neu} bezüglich der neuen Tabelle lauten:
$f_{\text{neu}}(A) = 2$, $f_{\text{neu}}(B) = 5$, $f_{\text{neu}}(C) = 0$, $f_{\text{neu}}(D) = 5$ und $f_{\text{neu}}(E) = 10$

	Inhalt	gelöscht	belegt	kollidiert	behandelt
0	A	-	X	X	-
1	B	-	X	-	-
2	C	-	X	-	-
3	D	-	X	-	-
4	E	X	X	X	-
5					
6					
7					
8					
9					
10					

Führen Sie ein Rehashing durch und verwenden Sie dabei das auf Seite 75 des Vorlesungsskripts beschriebene Verfahren. Geben Sie den Zustand der Tabelle nach jedem Kopierbefehl und den jeweiligen Inhalt von *Puffer* und *Hilfspuffer* an!

3. Doppel-Hashing (mittel) (schriftlich, 3 Punkte)

Gegeben sind eine Hashtabelle der Größe $m = 13$, die Hashfunktion $h(k) = k \bmod m$ und die Kollisionsstrategie Doppel-Hashing mit

$$h_i(k) = (h(k) + i \cdot (1 + (k \bmod 7))) \bmod m$$

In welcher Reihenfolge können die Schlüssel 7, 8, 23, 33 und 60 in die zunächst leere Hashtabelle eingefügt worden sein, wenn die Hashtabelle wie folgt gefüllt ist:

Speicherzelle i	0	1	2	3	4	5	6	7	8	9	10	11	12
Belegung	23					60		33	7		8		

Begründen Sie Ihre Antwort!

4. Levelorder I (mittel) (schriftlich, 4 Punkte)

Auf Aufgabenblatt 8 sollten Sie die Lösungen einiger Aufgaben als Ausgabe in Levelorder abgeben.

Schreiben Sie nun ein Ada-Programm, das die Knoteninhalte eines binären Baumes in Levelorder-Reihenfolge auf dem Bildschirm ausgibt.

Geben Sie im eClaus-System zusätzlich eine ausführliche Beschreibung des Algorithmus ab!

5. Levelorder II (mittel) (schriftlich, 2 Punkte)

Gibt es zu jeder beliebigen Folge von ganzen Zahlen, die als Levelorder-Ausgabe interpretiert wird, einen Suchbaum? Begründen Sie Ihre Antwort!

6. Sortieren (leicht) (schriftlich, 2 Punkte)

Vorausblickend auf Kapitel 10 des Skripts wurden verschiedene Sortierverfahren kurz vorgestellt und ihre Zeitkomplexitäten genannt. Alle diese Zeitkomplexitäten lagen dabei zwischen $O(n \cdot \log n)$ und $O(n^2)$.

Diskutieren Sie folgende Fragestellung: Kann es ein Sortierverfahren geben, welches nur $O(n)$ oder gar $O(\log n)$ Aufwand benötigt? Wenn ja, welche Idee könnte hinter einem solch schnellen Sortierverfahren stecken? Wenn nein, geben Sie Gründe hierfür an!