TECHNISCHE UNIVERSITÄT MÜNCHEN
FAKULTÄT FÜR INFORMATIK

Lehrstuhl für Effiziente Algorithmen
Grundlagen: Algorithmen und Datenstrukturen
Prof. Dr. Christian Scheideler, Dr. Stefan Schmid

SS 2008
Übungsblatt 5
14.05.08

Abgabe: 14.05. - 21.05.08 (nach der Vorlesung)

Übung Grundlagen: Algorithmen und Datenstrukturen

Aufgabe 5.1 [4 Punkte] Worst Case

Stellen Sie sich vor, Sie benötigen für eine „real time"-Anwendung eine Datenstruktur für unbeschränkte Arrays, bei der die Ausführungszeit für jede Methode im worst case konstant ist. Erstellen Sie eine solche Struktur (Pseudocode) mit den folgenden bekannten Operationen:

- `pushBack(e: Element)`
- `popBack()`

Orientieren Sie sich dabei an der Klasse UArray aus der Vorlesung. (Bedenken Sie, dass dort diese beiden Methoden im worst case lineare Laufzeit haben.)
Begründen Sie Ihre Lösung!

Hinweis: Speichern Sie die Elemente in bis zu zwei Arrays. Verschieben Sie die Elemente in ein größeres Array bevor eine erneute Änderung der Größe notwendig ist.

Aufgabe 5.2 [6 Punkte] Amortisierte Analyse eines Binärzählers

Wir betrachten eine nichtnegative ganze Zahl c, repräsentiert durch ein Array der Binärdarstellung, und eine Sequenz von m Inkrement- und Dekrement-Operationen (also $c + 1$ und $c - 1$). Zu Anfang ist $c = 0$.

a) Wie ist die Laufzeit einer Inkrement-Operation bzw. einer Dekrement-Operation im **schlechtesten** Fall in Abhängigkeit von m? Bedenken Sie, dass Sie pro Schritt nur ein Bit ändern können.

b) Beweisen Sie, dass die **amortisierten** Kosten der Inkrement-Operationen konstant sind, wenn keine Dekrement-Operationen auftreten.
Hinweis: Definieren Sie sich dazu (zum Beispiel) das Potential von c als Anzahl der Einsen in der Binärdarstellung von c.

c) Geben Sie eine Sequenz von m Operationen an, so dass die Kosten in $\Theta(m \log m)$ liegen.

Aufgabe 5.3 [3 Punkte] **While Schleifen**

Für zwei Zahlen $a, b \in \mathbb{N}$ hat der grösste gemeinsame Teiler (ggT) d von a und b die Eigenschaft, dass

1. d teilt a und d teilt b und
2. für alle Zahlen c mit dieser Eigenschaft gilt, dass $c \leq d$.

Der folgende Algorithmus berechnet den ggT von a und b:

$$x := a; \quad y := b;$$

$$\textbf{while } (y \neq 0) \textbf{ do } \{z := x \bmod y; x := y; y := z; \}$$

$$d := x$$

$$\textbf{return}(d);$$

Sei $n = \log(a + b)$. Zeigen Sie mit einem Potentialfunktion Argument, dass der Algorithmus eine Laufzeit von $O(n)$ hat.

Hinweis: Sei z_i der Wert von z zu Beginn des i-ten Schleifendurchlaufs. Stellen Sie eine geeignete Beziehung zwischen z_i und z_{i+2} her.