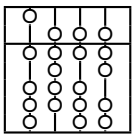
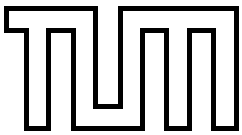


TECHNISCHE UNIVERSITÄT MÜNCHEN
FAKULTÄT FÜR INFORMATIK



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

SS 2008
Übungsblatt 4
07.05.08

Abgabe: 07.05. - 14.05.08 (nach der Vorlesung)

Übung Grundlagen: Algorithmen und Datenstrukturen

Aufgabe 4.1 [5 Punkte] Potential Methode

Betrachten Sie die Potential-Methode angewandt auf unbeschränkte Arrays für beliebige Werte α und β ($\alpha > \beta > 1$). Zeigen Sie, dass in jedem Schritt genug Token zur Verfügung stehen, wenn bei jedem Aufruf von `pushBack` $\beta/(\beta-1)$ Token, bei jedem Aufruf von `popBack` $\beta/(\alpha-\beta)$ Token einbezahlt werden.
Zeigen Sie dafür, dass bei jedem Aufruf von `reallocate` genügend Token vorrätig sind. Überlegen Sie sich dazu, dass `reallocate` immer mit dem Wert $\beta n'$ (für ein $n' \in \mathbb{N}$) aufgerufen wird und nach einem solchen Aufruf das Array $w = n'\beta$ Stellen hat, von denen n' belegt und $(\beta-1)n' = ((\beta-1)/\beta)w$ Stellen frei sind. `reallocate` wird erst wieder aufgerufen, falls $n = w$ oder $\alpha n \leq w$ ist.

Aufgabe 4.2 [4 Punkte] Sortiertes Feld mit Lazy Update

In dieser Aufgabe betrachten wir Operationen auf einem sortierten Feld $F = [0, ..., n-1]$. Wir nehmen an, dass dieses Feld zu jedem Zeitpunkt maximal n Elemente hat, und zu Beginn bereits sortiert ist. Auf F sollen nun folgende Operationen ausgeführt werden: $insert()$, $remove()$ und $find()$ (siehe Folie 3/59).

Nehmen Sie an, es gibt ein weiteres Feld $Op = [0, ..., \sqrt{n}-1]$ der Grösse $\sqrt{n}$. Operationen $insert()$, $remove()$ und $find()$ auf F werden *lazy bearbeitet*: solange es noch Platz hat im Feld Op werden sie einfach hinten in Op eingefügt statt das Feld F direkt zu bearbeiten – das Feld F bleibt also unverändert. Wir nehmen an, dass eine solche Einfügeoperation auf Op konstante Kosten $O(1)$ hat. Soblad das Op Feld aber voll ist, muss ein $update()$ ausgeführt werden, bei welchem alle in Op gespeicherten Operationen aufs Feld F übertragen werden und Op geleert wird. Natürlich soll das Feld danach wieder sortiert sein. Wir nehmen an, eine solche $update()$ Funktion habe Kosten n.

Zeigen Sie, dass der amortisierte Aufwand für die Operationen $insert()$, $remove()$ und $find()$ durch diese lazy Bearbeitung durch $O(\sqrt{n})$ beschränkt ist.

Aufgabe 4.3 [4 Punkte] **Bonusaufgabe: Implementation von UArrays**

Implementieren Sie eine Klasse UArray mit den folgenden Operationen (siehe Vorlesung Kapitel 3, Folie 11):

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

Bieten Sie ein Menu an (interaktive Useranfrage über Kommandozeile oder GUI), welches dem Benutzer erlaubt, wahlweise zwischen einem pushBack einer integer Zahl, einem popBack oder einer Feldausgabe zu wählen.