Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr/Dr. Sven Kosub
Stefan Eckhardt

Wintersemester 2004/05
Übungsblatt 3
22. November 2004

Effiziente Algorithmen und Datenstrukturen I

Abgabetermin: 15.11.2004 **vor** *der Vorlesung*

Aufgabe 1

a) Zeigen Sie, wie man mit Hilfe von zwei Stacks eine Queue simulieren kann. Beschreiben Sie dazu, wie Sie die Befehle DEQUEUE und ENQUEUE mit Hilfe der Befehle PUSH und POP auf den beiden Stacks implementieren können. Wie groß ist die Laufzeit Ihrer Operationen?

b) Zeigen Sie, wie man einen Stack mit Hilfe von zwei Queues simulieren kann. Beschreiben Sie dazu, wie Sie die Befehle PUSH und POP mit Hilfe der Befehle DEQUEUE und ENQUEUE auf den beiden Queues implementieren können. Wie groß ist die Laufzeit Ihrer Operationen?

Aufgabe 2

Gegeben sei eine Folge $(l_1, l_2, \ldots, l_n)$ mit n Elementen, die als einfach verkettete Liste L implementiert ist. Geben Sie eine *nicht-rekursive* Prozedur INVERTLIST(L) an, die die Liste invertiert: Nach Anwendung der Prozedur soll L also die Folge $(l_n, l_{n-1}, \ldots, l_2, l_1)$ repräsentieren. Ihre Prozedur sollte nur $O(1)$ zusätzlichen Platz benötigen.

Aufgabe 3

Geben Sie einen Algorithmus an, der k sortierte Listen zu einer sortierten Liste zusammensetzt. Die Gesamtzahl der Elemente in allen Listen sei dabei n. Ihr Algorithmus sollte die Laufzeit $O(n \log k)$ haben.

Aufgabe 4

Gegeben sei ein Array $A = [1..n]$ mit n unterschiedlichen Schlüsseln. Die folgende Prozedur RANDOMSEARCH soll verwendet werden, um einen Schlüssel in A zu suchen.

```
RandomSearch(A, k)
i := 1
while i ≤ n and A[i] < k
    Random j in{1, ..., n}
    if A[i] < A[j] and A[j] < k
        i := j
    i := i + 1
    if A[i] = k
        return i
return „Schlüssel nicht vorhanden“
```

Zeigen Sie, dass die erwartete Laufzeit für einen Aufruf von RANDOMSEARCH $O(\sqrt{n})$ ist.