

Übungen zur Vorlesung
Datenstrukturen und Algorithmen
SS 2004
Blatt 2

AUFGABE 1 (12 Punkte):

Wir betrachten den folgenden Sortieralgorithmus, genannt *Bubble-Sort*.

BUBBLE-SORT(A)

```
1 for  $i \leftarrow 1$  to  $length(A)$ 
2   do for  $j \leftarrow length(A)$  downto  $i + 1$ 
3     do if  $A[j] < A[j - 1]$ 
4       ▷Vertausche  $A[j], A[j - 1]$ 
5     then  $A[j] \leftrightarrow A[j - 1]$ 
```

1. Formulieren und beweisen Sie zunächst eine Invariante für die **for**-Schleife in den Zeilen 2-5. Formulieren und beweisen Sie dann eine Invariante für die **for**-Schleife in den Zeilen 1-5. Beweisen Sie die Korrektheit von Bubble-Sort mit Hilfe dieser Invarianten. (6 Punkte)
2. Analysieren Sie die worst-case Laufzeit von Bubble-Sort. (6 Punkte)

AUFGABE 2 (8 Punkte):

Wir betrachten wieder das *Suchproblem*. Wir nehmen aber an, dass die Eingabefolge A bereits sortiert ist. Also

Eingabe: Eine sortierte Folge von n Zahlen (a_1, \dots, a_n) , die in einem Array A gespeichert ist, und eine weitere Zahl v .

Ausgabe: Ein Index i , so dass $v = A[i]$ oder ein spezieller Wert NIL, falls v nicht in der Folge A auftaucht.

Schreiben Sie Pseudocode für die so genannte *binäre Suche*: Bei der binären Suche wird zunächst geprüft, ob v in der oberen oder unteren Hälfte des Arrays auftauchen kann. Dann wird rekursiv in der passenden Hälfte des Arrays weitergesucht.

1. Schreiben Sie Pseudocode für die binäre Suche.
2. Analysieren Sie die worst-case Laufzeit Ihres Algorithmus.
Hinweis: Die Laufzeit sollte von der Form $c \log(n)$ sein.