

Übungsblatt 3

keine schriftlichen Aufgaben

Aufgabe 8

mündlich

Bubble-Sort sortiert eine Zahlenfolge $a_1, \dots, a_n$ durch wiederholtes Vertauschen von benachbarten Folgengliedern:

Bubble-Sort

Input: Eine Zahlenfolge $a_1, \dots, a_n$
for $i := n - 1$ **downto** 1 **do**
 for $j := 1$ **to** i **do**
 if $a_j > a_{j+1}$ **then** vertausche(a_j , a_{j+1})
Output: $a_1, \dots, a_n$

- (a) Wie verarbeitet **Bubble-Sort** die Eingabefolge 3, 6, 1, 7, 9, 2, 4, 8?
- (b) Finden und beweisen Sie eine geeignete Invariante für die innere **for**-Schleife.
- (c) Benutzen Sie die Schleifeninvariante aus (b) für den Nachweis einer geeigneten Invariante für die äußere **for**-Schleife.
- (d) Beweisen Sie die Korrektheit von **Bubble-Sort** mithilfe der Invariante aus (c).
- (e) Bestimmen Sie die Anzahl an Vergleichen, die **Bubble-Sort** im besten und im schlechtesten Fall benötigt.
- (f) Verbessern Sie die bestcase-Komplexität von **Bubble-Sort**, indem Sie die äußere **for**-Schleife durch eine **repeat**-Schleife ersetzen, die für möglichst wenige Werte von i durchlaufen wird und insbesondere abbricht, sobald die Folge sortiert ist.

Aufgabe 9

mündlich, optional

Implementieren Sie **MergeSort** als nichtrekursives »in place«-Sortierverfahren.

Hinweis: Die zu sortierende Zahlenfolge soll durch einen Zeiger auf eine einfach verkettete Liste L übergeben werden. Bei jedem Schleifendurchlauf soll der Algorithmus die Liste L zuerst so in zwei Listen A und B zerlegen, dass sortierte Teilstücke maximaler Länge (so genannte *Runs*) zusammenbleiben und abwechselnd auf die beiden Listen A und B verteilt werden. Im zweiten Teil der Schleife sollen die Listen A und B wieder zu einer Liste gemischt werden, so dass sich die Anzahl der Runs bei jedem Schleifendurchlauf halbiert.

Aufgabe 10

mündlich

Die Prozedur **Merge** benötigt im schlechtesten Fall $n + m - 1$ Vergleiche, um zwei sortierte Zahlenfolgen $a_1 \leq \dots \leq a_n$ und $b_1 \leq \dots \leq b_m$ zu einer sortierten Folge $c_1 \leq \dots \leq c_{n+m}$ zusammenzufügen.

- (a) Zeigen Sie, dass es keine Prozedur gibt, die dies im Fall $n = m$ mit weniger als $2n - 1$ Vergleichen schafft.
- (b) Überlegen Sie sich ein Verfahren, das im Fall $m \ll n$ (z.B. $m \approx \sqrt{n}$) mit deutlich weniger Vergleichen auskommt.

Aufgabe 11

mündlich

Konstruieren Sie einen vergleichsbasierten Algorithmus, der im schlechtesten Fall eine möglichst geringe Anzahl $V(n)$ an Vergleichen benötigt, um

- (a) das Maximum und das Minimum,
- (b) das größte und zweitgrößte Element

einer Folge von n Zahlen zu finden.

Hinweis: Für (a) sind $\lceil 3n/2 \rceil - 2$ und für (b) sind $n + \lceil \log_2 n \rceil - 2$ Vergleiche optimal.

Aufgabe 12**6 Punkte**

Bestimmen Sie die minimale Anzahl an Vergleichen, die ein vergleichsbasierter Algorithmus im besten Fall benötigt, um

- (a) eine Folge von n Zahlen zu sortieren, *(mündlich)*
- (b) den Median einer Folge von n Zahlen zu finden, *(2 Punkte)*
- (c) das Maximum einer Folge von n Zahlen zu finden, *(2 Punkte)*
- (d) das Maximum und das Minimum einer Folge von n Zahlen zu finden. *(2 Punkte)*