Algorithmen und
Berechnungskomplexität I
Wintersemester 2016/17

Abgabe: 03.11.2016, 12.30 Uhr

Prof. Dr. Heiko Röglin
Clemens Rösner
Institut für Informatik

Übungsblatt 2

Aufgabe 2.1: (2+3 Punkte)

Wir betrachten die Rekursionsgleichung $T(n) = 4 \cdot T(n/2) + n$ für Zweierpotenzen n mit $T(1) = 1$.

(a) Zeigen Sie, dass man die Ungleichung

$$T(n) \leq cn^2 \tag{1}$$

für keine Konstante c direkt mittels Induktion zeigen kann.

(b) Beweisen Sie Ungleichung (1) für eine geeignete Konstante c, indem Sie die stärkere Ungleichung

$$T(n) \leq cn^2 - f(n) \tag{2}$$

für eine geeignete Funktion $f(n) \geq 0$ induktiv zeigen.

Aufgabe 2.2: (3 Punkte)

Wir betrachten folgenden rekursiven Sortieralgorithmus:

```
Sort(A, n)
  1. if n ≥ 2 then
  2.    Rufe Sort(A, n − 1) auf.
  3.    Schiebe A[n] nach links, bis es an der richtigen Stelle steht.
  4. end if
```

Geben Sie eine Rekursionsformel für die Worst-Case Laufzeit des obigen Algorithmus an (eine Abschätzung nach oben genügt) und lösen Sie diese.

Aufgabe 2.3: (6 Punkte)

Wir wollen die Laufzeiten von *Insertionsort* und *Mergesort* experimentell miteinander vergleichen. Erzeugen Sie dazu für $n = 1, \ldots, 500$ n-mal hintereinander ein Feld mit n zufälligen Einträgen und führen Sie *Insertionsort* und *Mergesort* auf diesem Feld aus. Bestimmen Sie für jedes n die durchschnittliche Laufzeit beider Sortierverfahren und stellen Sie diese als Graphen in Abhängigkeit von n dar.

Aufgabe 2.4: (3+3+4 Punkte)

Wir betrachten ein Feld A mit n paarweise verschiedenen Einträgen.

(a) Geben Sie einen Algorithmus an, der ein Feld A mit n Elementen und eine beliebige Zahl $k \in \{1, \ldots, n\}$ als Eingabe erhält und das k-t kleinste Element von A in Linearzeit ermittelt.

Hinweis: Verwenden Sie eine Divide-and-Conquer-Strategie. Sie können davon ausgehen, dass der *Median* eines Feldes A, also das M-t kleinste Element von A mit $M = \lceil n/2 \rceil$, in Linearzeit bestimmt werden kann.

(b) Beweisen Sie die Korrektheit Ihres Algorithmus.

Hinweis: Verwenden Sie Induktion über die Größe des Feldes.

(c) Zeigen Sie, dass die Laufzeit Ihres Algorithmus linear ist. Sie können sich dabei auf den Fall einschränken, dass n eine Zweierpotenz ist.

Aufgabe 2.5: (6 Zusatzpunkte)

Sei A ein Feld mit n Einträgen $A[1], \ldots, A[n]$. Die Anzahl $\chi(A)$ der *Inversionen* oder *Fehlstellungen* von A ist definiert als

$$\chi(A) = |\{(i, j) : 1 \leq i < j \leq n \text{ und } A[i] > A[j]\}| .$$

Geben Sie einen Algorithmus an, der für ein Feld A mit n Einträgen die Anzahl $\chi(A)$ der Inversionen in Zeit $O(n \log n)$ bestimmt.