

5. Übung:

Algorithmen und Datenstrukturen

Sommersemester 2009

27. Mai 2009

Abgabe bis Montag, 15. Juni 2009, 10:00 im Briefkasten vor H426

Aufgabe 5.1:

(Punkte 4)

Konstruieren Sie Eingaben für QUICKSORT, so dass die Laufzeit

- a) $O(n \log n)$ ist.
- b) $\Omega(n^2)$ ist, aber die Eingabe weder aufsteigend noch absteigend sortiert ist.

Begründen Sie das jeweilige Laufzeitverhalten.

Aufgabe 5.2:

(Punkte 4)

Betrachten Sie die Variante von Quicksort, bei der jeweils das mittlere Element des zu sortierenden Teilfeldes als Pivot-Element ausgewählt wird. Konstruieren Sie eine Eingabe, die möglichst viele Vergleiche erzwingt. Wieviele Vergleiche sind möglich ?

Aufgabe 5.3:

(Punkte 5)

Entwickeln Sie einen Algorithmus $\text{SELECT}(A, k)$, der für ein Feld $A[1..n]$ von Zahlen, das k -t größte Element berechnet. Kann man das Problem lösen ohne das Feld zu sortieren ? Hinweis: Betrachten Sie folgende Idee für einen rekursiven Algorithmus. Ordne das Feld A so um, dass alle Zahlen, die kleiner oder gleich $A[1]$ sind, vor den restlichen Zahlen stehen. Wende dann denselben Algorithmus (rekursiv) auf den ersten oder den zweiten Abschnitt an.

Aufgabe 5.4:

(Punkte 6)

Entwickeln Sie eine *stabile* Variante von Counting-Sort, die im letzten Schritt das Eingabefeld von vorne nach hinten und nicht (wie in der Vorlesung) von hinten nach vorne liest. Hinweis: Überlegen Sie sich, wie die Definition der Werte im Feld C geändert werden muss.