



Übungsblatt 5

Abgabe via Moodle.
Deadline Fr. 9ter Juni

Aufgabe 1 (Algorithmendesign, 4 + 6 Punkte)

Gegeben sei eine sortierte Liste mit n Elementen. Nun haben Sie $k < n$ neue unsortierte Elemente in einem Array, die in die sortierte Liste eingefügt werden sollen, so dass die Liste nach dem Einfügen wieder sortiert ist.

1. *Warmup*: Geben Sie einen Algorithmus an, der Laufzeit $O(kn)$ hat und das Problem löst.
2. Geben Sie nun einen Algorithmus an, der (erwartete) Laufzeit $O(k \log k + n)$ hat und das Problem löst.

Aufgabe 2 (Anwendungsproblem, 6 + 2 Punkte)

In einer Fabrik werden n Waren pro Monat hergestellt, die zu unterschiedlichen Zeitpunkten von den Lieferanten abgeholt werden. In der Lagerhalle der Fabrik gibt es k Lagerplätze für die produzierten Waren. Der Fabrikleiter muss nun den Produktionsplan kontrollieren, ob genügend Lagerplätze vorhanden sind, um die entsprechenden Waren von Produktionsdatum bis Abholdatum zu lagern. Eine Produktionseinheit besteht also im Plan aus Produktionsdatum und Abholdatum. Der Manager möchte wissen, ob genug Plätze vorhanden sind um die Anforderungen zu erfüllen. Die Lieferanten kommen immer zu Beginn des Tages um die Ware abzuholen, also bevor Waren dieses Tages gelagert werden müssen.

1. Entwerfen Sie einen Algorithmus, der dieses Problem in höchstens $O(n \log n)$ Zeit löst.

Hinweis. Betrachten Sie die Menge aller Produktions- und Abholdaten. Sortieren Sie diese Menge und verarbeiten Sie sie dann in sortierter Reihenfolge.

2. Begründen Sie kurz, warum Ihr Algorithmus das gewünschte Laufzeitverhalten aufweist.

Aufgabe 3 (Sortieren, 6 Punkte)

Sortieren Sie die Ziffern Ihrer Matrikelnummer.

1. Benutzen Sie Insertionsort. Geben Sie den Zustand des Feldes nach jedem der 6 Insert-Schritte an.
2. Benutzen Sie Mergesort. Verwenden Sie das Schema aus dem Beispiel der Vorlesung (Folie 213, Stand 24.4., Überschrift "Mischen").
3. Benutzen Sie Quicksort. Verwenden Sie das Schema aus dem Beispiel der Vorlesung (Folie 235, Stand 24.4., Überschrift "Beispiel: Rekursion"). Als Pivot soll das erste Element verwendet werden.

Aufgabe P5 (*Algorithmen-design-praktisch, optional*)

Für die praktischen Übungen verwenden wir die Plattform www.hackerrank.com. Hier müssen Sie sich registrieren um an den Übungen teilzunehmen. Unter dem Link

<https://www.hackerrank.com/adsi-2023>

finden die praktischen Übungen in der Form eines Programmierwettbewerbs statt.

In der fünften Challenge geht es um Aufgabe 1 dieses Übungsblattes.

Gegeben ist eine sortierte, einfach verkettete Liste mit n Elementen. Nun haben Sie k neue unsortierte Elemente in einem Array, welche in die sortierte Liste eingefügt werden sollen, sodass die Liste nach dem Einfügen wieder sortiert ist.

Die Aufgabe ist es nun die beiden Algorithmen Ihrer Lösung aus Aufgabe 1 zu implementieren. Sie können in der Challenge die Laufzeiten der beiden Algorithmen vergleichen. Dazu müssen die in der main Funktion markierten Zeilen (164 - 184) auskommentiert werden.

Eine genauere Beschreibung, sowie ein Beispiel finden Sie auf HackerRank.