

Prof. Dr. Hans-Peter Kriegel  
Thomas Bernecker, Tobias Emrich

Übungen zur Vorlesung  
**Effiziente Algorithmen**

**Hausaufgabe 9.1: Heap-Sort (6 Punkte)**

Sortieren Sie die folgenden Zahlen mit Heap-Sort: 14, 93, 77, 64, 36, 88, 31, 77, 34, 11.

Zeichnen Sie dabei den Heap beim Aufbau für jeden Level und nach jedem Absenken eines Schlüssels, sowie bei den Sortierschritten nach jedem Absenken. Verwenden Sie dabei einen Baum als Darstellung für den Heap.

**Hausaufgabe 9.2: Sortierverfahren mit linearer Zeitkomplexität (3 + 1 Punkte)**

Es seien  $n$  Elemente  $a_1, a_2, \dots, a_n \in M$  zu sortieren, wobei  $M$  eine endliche Menge sei, auf der eine vollständige Ordnungsrelation „ $\leq$ “ definiert sei. Ohne Beschränkung der Allgemeinheit nehmen wir an, daß es sich bei  $M$  um die Menge der natürlichen Zahlen von 1 bis  $m$  handelt, also  $M = \{1, 2, \dots, m\}$  und  $m, n \in \mathbb{N}$ ,  $m = cn$ , mit einer Konstante  $c \in \mathbb{Q}$ ,  $c > 0$ .

- Geben Sie einen Algorithmus in Java-Notation zur Sortierung der Elemente  $a_1, a_2, \dots, a_n$  an, der in der Komplexitätsklasse  $O(n)$  liegt. Erläutern Sie Ihre Lösung.
- Erklären Sie die Tatsache, daß ein linearer Aufwand möglich ist, obwohl in der Vorlesung  $O(n \cdot \log n)$  als typische Komplexitätsklasse für gute allgemeine Sortierverfahren herausgestellt wurde.

**Hausaufgabe 9.3: Radix-Sort (3 + 1 Punkte)**

- Sortieren Sie die angegebene Zahlenfolge mit Hilfe des Radix-Sort Verfahrens. Geben Sie dabei die Belegung der einzelnen Buckets für jede Phase an.

234, 479, 321, 128, 111, 9, 66, 999, 989, 123,  
130, 142, 0, 318, 732, 38, 259, 300, 748, 200.

- Nehmen Sie an, Sie hätten die folgenden Zahlen zu sortieren:

1234567834, 4348280976, 9192939467, 9192939067

Würden Sie das Radix-Sort Verfahren verwenden oder ein Verfahren, das nur mit Schlüsselvergleichen arbeitet? Begründen Sie Ihre Aussage.