

## Übung zur Vorlesung Algorithmen und Datenstrukturen

### Aufgabe T1

Ordnen Sie die folgenden Funktionen in der Reihenfolge ihres asymptotischen Wachstums (ohne lange Nachzudenken). Mit  $\log n$  bezeichnen wir den Logarithmus zur Basis 2.

- $\log(n)$
- $\log(n^5)/\log \log(n)$
- $n^{\log \log n}$
- $\log(n)^{\log n}$
- $n^2$
- $n \log n$
- $2^n$

### Aufgabe T2

Für welche Funktionen  $f: \mathbf{N} \rightarrow \mathbf{N}$  gilt  $f(\lceil O(1) \rceil) = O(1)$ ? Beweisen Sie Ihre Behauptung mithilfe der Definition der  $O$ -Notation.

### Aufgabe T3

Entwerfen Sie einen einfachen Algorithmus, der zwei doppelt verkettete Listen aneinanderhängt und auf diese Weise eine neue doppelt verkettete Liste erzeugt. Können Sie dabei auf bedingte Verzweigungen verzichten?

Schreiben Sie den Algorithmus in Pseudocode oder in einer Programmiersprache nieder.

### Aufgabe H1 (10 Punkte)

Gegeben sei ein Array  $a[0], \dots, a[n-1]$ , welches Zahlen in aufsteigend sortierter Reihenfolge enthält. Entwerfen Sie einen effizienten Algorithmus, der die Anzahl der Zahlen in diesem Array bestimmt, welche sich in einem Intervall  $[l, r]$  befinden, wobei  $l$  und  $r$  ebenfalls zwei Zahlen sind.

Die Laufzeit ihres Algorithmus sollte  $O(\log n)$  betragen.

Erläutern Sie die Arbeitsweise des Algorithmus und begründen Sie, warum er schnell ist. Beachten Sie insbesondere den Fall, wenn das Array auch die Zahlen  $l$  und  $r$  mehrfach enthält. Auch dann muß das Ergebnis korrekt sein.

Implementieren Sie Ihren Algorithmus in einer geeigneten Sprache und lassen Sie ihn auf einigen aussagekräftigen Beispielen laufen. Verwenden Sie dabei auch Fälle mit  $n = 10000000$ . Überlegen Sie sich ein Verfahren, wie sie die Laufzeit ihrer Suche in diesem Falle möglichst gut messen können und geben Sie die dabei gefundene Zeit an.