Wintersemester 2008/09 | Forschungsinstitut für Diskrete Mathematik
Prof. Dr. S. Hougardy | Universität Bonn
Dr. U. Brenner
C. Mohr

Algorithmische Mathematik I

6. Übung

1. Beweisen oder widerlegen Sie die folgenden Aussagen:

   (a) Es sei $f(x) = \sum_{i=0}^{n} a_i \cdot x^i$ mit $a_i \in \mathbb{R}$. Dann gilt $f(x) = \mathcal{O}(x^n)$.

   (b) Seien $f(x) = \mathcal{O}(g(x))$ und $g(x) = \mathcal{O}(h(x))$. Dann gilt $f(x) = \mathcal{O}(h(x))$.

   (c) Sei $f(x) = \mathcal{O}(g(x))$. Dann gilt $e^{f(x)} = \mathcal{O}(e^{g(x)})$.

   (d) $o(f(x) \cdot g(x)) = f(x) \cdot o(g(x))$.

   (e) Aus $g(x) = o(f(x))$ folgt $f(x) + g(x) = \Theta(f(x))$.

   (f) $n^{\left(\frac{4}{\log n^{\varepsilon}}\right)^2} = \Omega(n)$, wobei $\varepsilon > 0$ eine beliebige Konstante sei.

   (g) $x^6 = \mathcal{O}(2^x)$.

   (h) $f(x) = \mathcal{O}(g(x)) \quad \Leftrightarrow \quad \exists c > 0$ mit $\lim_{x\to\infty} \frac{f(x)}{g(x)} \leq c$.

   (i) $x \log(x) = \mathcal{O}(x^{1+\varepsilon})$ für alle $\varepsilon > 0$. (11 Punkte)

2. Schreiben Sie ein C++-Programm, das n reelle Zahlen $a_1, \ldots, a_n$ in alternierender Reihenfolge sortiert, das heißt nach geeigneter Permutation $\pi : \{1, \ldots n\} \to \{1, \ldots n\}$ soll gelten

$$a_{\pi(1)} \leq a_{\pi(2)} \geq a_{\pi(3)} \leq a_{\pi(4)} \geq a_{\pi(5)} \leq \ldots$$

   (10 Punkte)

3. Betrachten Sie eine Variante von Merge-Sort, die entsteht, wenn die zu sortierende Menge M nicht nur in zwei Teilmengen sondern in $k \geq 2$ möglichst gleich große Teilmengen zerlegt wird. Bestimmen Sie die Laufzeit dieses Verfahrens. (10 Punkte)

4. Bestimmen Sie die maximale Zahl der Vergleiche, die bei

   (a) Merge-Sort

   (b) Quick-Sort

   benötigt werden, um fünf Elemente $a_1, \ldots, a_5$ zu sortieren. Vergleichen Sie die Resultate mit der unteren Schranke für das Sortieren von fünf Elementen. (9 Punkte)

Abgabe: Mittwoch, den 26.11.2008, **vor** der Vorlesung.