Prof. Dr. Hans-Jörg Kreowski, Renate Klempien-Hinrichs
Studiengang Informatik
MZH 3260, Tel.: 2956, 3697 (Sekr.), Fax: 4322
E-Mail: {kreo, rena}@informatik.uni-bremen.de

23. Mai 2000
Aufgaben 4

Theoretische Informatik 1

4. Übungsblatt

Von den folgenden Aufgaben sollen die am Rand mit Prozentzahlen versehenen bearbeitet werden. Dabei kann wahlweise Aufgabe 1 oder Aufgabe 3(h) weggelassen werden.

1. Minimaler Aufwand

   Bisher wurde der Aufwand von Operationen genau ermittelt oder nach oben abgeschätzt. Hier sollen nun untere Schranken für den Aufwand untersucht werden.

   (a) Betrachte die Spezifikation **mergesort** aus Abschnitt 3.7 des Skripts. Gib für alle $m, n \geq 1$ Eingabewörter $u, v \in \mathbb{N}^*$ der Längen $\mathit{length}(u) = m$ und $\mathit{length}(v) = n$ an, so dass die Auswertung von $\mathit{merge}(u, v)$ mindestens $m + n$ Schritte braucht. (10%)

   (b) Betrachte die Spezifikation **quicksort** vom 3. Übungsblatt. Zeige, dass die Auswertung von $\mathit{qsort}(w)$ für Eingabewörter w der Länge n, die falsch herum sortiert sind, mindestens $n^2 + 3n$ Schritte braucht. Gilt das auch für Eingabewörter, die richtig herum sortiert sind? (20%)

2. Matrizenmultiplikation

   Der klassische Algorithmus zur Multiplikation von (n, n)-Matrizen benötigt n^3 Multiplikationen und $n^3 - n^2$ Additionen. Dagegen verwendet der Algorithmus von Winograd $\frac{1}{2}n^3 + n^2$ Multiplikationen und $\frac{3}{2}n^3 + 2n^2 - 2n$ Additionen.

   (a) Für welche $n \in \mathbb{N}$ arbeitet der Algorithmus von Winograd effizienter, wenn eine Multiplikation gegenüber einer Addition fünfmal soviel Zeit benötigt? (10%)

   (b) Zeige, dass der Algorithmus von Winograd zur Matrizenmultiplikation nicht effizienter ist als der klassische, wenn Addition und Multiplikation gleich viel Zeit benötigen. (10%)

3. Aufwandsklassen

   Für Funktionen $f, g \colon \mathbb{N} \to \mathbb{N}$ gilt $g \in O(f)$ genau dann, wenn natürliche Zahlen c, n_0 existieren, so dass $g(n) \leq c \cdot f(n)$ für alle $n \geq n_0$.

   (a) Zeige, dass $O(g) \subseteq O(f)$ gilt, falls $g \in O(f)$. (10%)

   (b) Zeige, dass $O(g) = O(f)$ genau dann, wenn sowohl $g \in O(f)$ als auch $f \in O(g)$ gilt.

   (c) Es gilt $O(1) \subseteq O(\mathrm{ld}\, n) \subseteq O(n) \subseteq O(n\, \mathrm{ld}\, n) \subseteq O(n^2) \subseteq O(2^n)$.
   Weise $O(n^2) \subseteq O(2^n)$ nach. (20%)

   (d) Alle Inklusionen in (c) sind echt.
   Weise dies nach für $O(\mathrm{ld}\, n) \subseteq O(n)$. (20%)

   (e) Sei $g \in O(f)$ und $h(n) = f(n) + g(n)$. Zeige, dass dann $O(h) = O(f)$ gilt.

   (f) Für alle $k, l \in \mathbb{N}$ gilt $n^k \in O(n^l)$ genau dann, wenn $k \leq l$.
   Also gilt $O(n^k) \subsetneqq O(n^l)$ genau dann, wenn $k < l$.

   (g) Wann gilt demnach $O(p) = O(q)$ für Polynome p und q? Warum?

   (h) Wie verhalten sich die Klassen $O(n!)$ und $O(2^n)$ zueinander (d.h., ist eine von beiden echt in der anderen enthalten, sind beide gleich, weder noch)? Weise die Korrektheit der Antwort nach. (30%)

   (i) Wie verhalten sich die Klassen $O(n^n)$ und $O(n!)$ zueinander?

   (j) Gilt für alle f, g entweder $g \in O(f)$ oder $f \in O(g)$?

Abgabe: 8. Juni 2000 in der Vorlesung.