Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr/Dr. Sven Kosub
Stefan Eckhardt

Wintersemester 2004/05
Übungsblatt 1
12. November 2004

Effiziente Algorithmen und Datenstrukturen I

Abgabetermin: 29.10.2004 **vor** *der Vorlesung*

Aufgabe 1

(a) Beweisen Sie $\log(n!) = \Theta(n \cdot \log n)$. Verwenden Sie dazu *nicht* die Stirlingsche Formel.

(b) Zeigen Sie $n! = o(n^n)$.

(c) Gilt $\log^k n = o(n^{\frac{1}{2^k}})$ für alle $k \in \mathbb{N}$? Begründen Sie Ihre Antwort.

(d) Wächst $\lceil \log n \rceil!$ asymptotisch wie ein Polynom? Wie verhält es sich mit $\lceil \log \log n \rceil!$? Begründen Sie Ihre Antwort.

Aufgabe 2

Die n-te *harmonische Zahl* H_n ist definiert als $H_n =_{\text{def}} \sum_{k=1}^{n} \frac{1}{k}$.

(a) Zeigen Sie, dass $1 + \frac{n}{2} \leq H_{2^n} \leq 1 + n$ für alle $n \in \mathbb{N}$ gilt.

(b) Drücken Sie die Summe $1 + \frac{1}{3} + \frac{1}{5} + \ldots + \frac{1}{2n-1}$ durch harmonische Zahlen aus.

(c) Drücken Sie folgende Summen durch harmonische Zahlen aus

- $\displaystyle\sum_{k=1}^{n} H_k$ und
- $\displaystyle\sum_{k=1}^{n} \binom{k}{m} H_k$ für $m \in \mathbb{N}$.

Aufgabe 3

Geben Sie eine Funktion $f : \mathbb{N} \to \mathbb{R}$ mit $T(n) = \Theta(f(n))$ an, wobei T durch die folgende Rekursion gegeben ist:

$T(x) = c$ für alle $x \leq 1$.

$T(x) = T(\alpha x) + T((1-\alpha)x) + c \cdot x$ für $x > 1$.

Hierbei sind $0 < \alpha < 1$ und $c > 0$.

Aufgabe 4

Zeigen Sie, dass es keinen Algorithmus gibt, der das Maximum von n natürlichen Zahlen bestimmt und dafür stets höchstens $n-2$ Vergleiche benötigt.