

3. Übungsblatt

Abgabe bis zum 22.11.05

www.math.tu-berlin.de/Vorlesungen/WS05/Programmiermethoden

1. Aufgabe

(5 Punkte)

Zeigen Sie, dass für $f, g : \mathbb{N} \rightarrow \mathbb{R}$ gilt:

$$\lim_{n \rightarrow \infty} \frac{|f(n)|}{|g(n)|} = 0 \iff \left\{ \forall \varepsilon \in \mathbb{R}^+ \exists n_0 \in \mathbb{N} : |f(n)| \leq \varepsilon |g(n)| \forall n \geq n_0 \right\} \implies f = o(g).$$

Für die erste Äquivalenz sei $\lim_{n \rightarrow \infty} g(n) \neq 0$.

2. Aufgabe

(1+2 Punkte)

- Zeigen Sie: Aus $f = o(g)$ folgt $g \neq o(f)$.
- Sei $f(n) = n^2 \sqrt{n}$, $g(n) = 5n^2$. Welche der folgenden Aussagen ist richtig?
(i) $f = o(g)$, (ii) $g = o(f)$. Begründen Sie Ihre Antwort!

3. Aufgabe

(3+2 Punkte)

- Schreiben Sie einen Algorithmus, der eine in b -adischer Darstellung gegebene natürliche Zahl in das Dezimalsystem umwandelt. D.h.:
Gegeben sei: $b \in \mathbb{N}, b > 1$ und $(z_{n-1} \dots z_0)_b, z_i \in \Sigma_b$.
Gesucht ist: $z \in \mathbb{N}$, also $\bar{z}_i \in \Sigma_{10}$ mit $(\bar{z}_{k-1} \dots \bar{z}_0)_{10} = (z_{n-1} \dots z_0)_b$.
- Wieviele Additionen und Multiplikationen braucht der Algorithmus?

4. Aufgabe

(2+1+2+2 Punkte)

Eine ganze Zahl $z \in \mathbb{Z}$ kann in der folgenden Form dargestellt werden:

$$z = \sum_{i=0}^{k-1} z_i 2^i - 2^{k-1}, \quad z_i \in \Sigma_2 = \{0, 1\}, i = 0, \dots, k-1, \quad k \in \mathbb{N}.$$

Abgespeichert wird dann die Bitsequenz $(z_{k-1} \dots z_0)$.

- Was ist für gegebenes $k \in \mathbb{N}$ der *darstellbare Zahlenbereich*, d.h. was ist die kleinste, was die größte darstellbare Zahl?
- Durch welche Bitsequenz wird die Zahl $(11)_{10}$ für $k = 5$ dargestellt?
- Lösen Sie zuerst die Programmieraufgabe 3a) s.u.
Was ist der darstellbare Zahlenbereich einer `short`-Variable in C++?
- Kommentieren bzw. begründen Sie die Ergebnisse der Programmieraufgabe 3 b) s.u.

3. Programmieraufgabe

(Vorführen bis zum 22.11.05)

In C++ gibt es die ganzzahligen Datentypen `short`, `int`, `long` und `long long`.

- Schreiben Sie ein Programm, das den Speicherbedarf in Byte (1 Byte = 8 Bit) dieser vier Datentypen in Tabellenform ausgibt. Benutzen Sie den Befehl `sizeof()`.
- Schreiben Sie ein Programm und finden Sie heraus, was passiert, wenn man
 - zu der größten als `short` darstellbaren Zahl 1 hinzuaddiert.
 - von der kleinsten als `short` darstellbaren Zahl 1 abzieht.