

Aufgabenblatt 1

Aufgabe 1, Sort Geschwindigkeit (1P)

Angenommen ein merge sort und ein insertion sort Algorithmus laufen auf dem gleichen Rechner. Für einen Input der Grösse n braucht der insertion sort $8 \cdot n^2$ Schritte, der merge sort benötigt für den gleichen Input $64 \cdot n \cdot \lg n$ Schritte. Für welche Werte von n ist welcher Algorithmus schneller?

Aufgabe 2, Vergleich von Laufzeiten (3P)

Bestimmen Sie (für jede Funktion $f(n)$ und für jede Zeit t in der Tabelle) die maximale Grösse n eines Problemes, das in der Zeit t gelöst werden kann, wenn der Algorithmus dazu $f(n)$ Mikrosekunden braucht.

	1 Sek.	1 Min.	1 Std.	1 Tag	1 Monat	1 Jahr
$\lg n$						
n						
n^2						
2^n						
$n!$						

Aufgabe 3, Funktionen (6P)

Wir schreiben $f = O(g)$ für zwei Funktionen f und g , falls

$$\exists c, n_0 > 0 : \forall n \geq n_0 : 0 \leq f(n) \leq c \cdot g(n).$$

Gegeben sind die folgenden Funktionen:

$$\begin{array}{lll}
 f_1(n) = 2^n & f_4(n) = n! & f_7(n) = \log_2 \sqrt{\log_2(\log_2 n)} \\
 f_2(n) = n^{1/n} & f_5(n) = \log_2(\log_2(n^2)) & f_8(n) = n^{\sqrt{n}} \\
 f_3(n) = \sqrt[5]{n} & f_6(n) = \pi^{\sqrt{5}n} & f_9(n) = \sum_{i=1}^n i
 \end{array}$$

Bringen Sie die Funktionen in eine Reihenfolge $f_{i_1} < f_{i_2} < \dots < f_{i_9}$, so dass gilt: $f_{i_j} = O(f_{i_{j+1}})$.

Begründen Sie Ihre Antwort, indem Sie für alle $1 \leq j \leq 8$ zeigen, dass

$$\lim_{n \rightarrow \infty} \frac{f_{i_j}(n)}{f_{i_{j+1}}(n)} < \infty.$$