Theoretische Informatik – Übung 12

SS 2019
Jochen Schmidt

Technische Hochschule Rosenheim

Folgende Aufgaben bitte vor der Übungsstunde zu Hause lösen:

Aufgabe 1

Bestimmen Sie die Komplexität der folgenden Codeausschnitte in O-Notation:

```
a) for( int i = n; i > 0; i /= 2 )
   {
     for( int j = 1; j < n; j *= 2 )
     {
       for( int k = 0; k < n; k += 2 )
       {

          ... // konstante Anzahl Operationen
        }
      }
   }
```

$O((\log_2 n)$

$O(\log_2 n)$

$O(n)$ / $O(\frac{n}{2})$

$O(1)$

$\log_2 n * \log_2 n * n * 1 = n(\log_2 n)^2$

```
b) for( int bound = 1; bound <= n; bound *= 2 )
   {
     for( int i = 0; i < bound; i++ )
     {
       for( int j = 0; j < n; j += 2 )
       {
         ... // konstante Anzahl Operationen
       }
       for( int j = 1; j < n; j *= 2 )
       {
         ... // konstante Anzahl Operationen
       }
     }
   }
```

$O(\log_2 n)$

$O(n)$

$O(n)$

$O(1)$

$O(\log_2 n)$

$O(1)$

$O(n + \log_2 n)$
$\Rightarrow O(n)$

$O(n^2 \cdot \log_2 n)$

Aufgabe 2

Zeigen Sie unter Verwendung der Definition der O-Notation, dass $T(n) = a_0 + a_1 n + a_2 n^2 + a_3 n^3 = O(n^3)$.
Hinweis: Berechnen Sie eine Konstante c und einen Schwellwert n_0, so dass $cn^3 \geq T(n)$ für $n \geq n_0$.

Aufgabe 3

Gegeben sind zwei Algorithmen A und B, die $T_A(n) = 5n \log_{10} n$ bzw. $T_B(n) = 25n$ Mikrosekunden für ein Problem der Größe n benötigen.

a) Welcher Algorithmus ist im Sinne der O-Notation der bessere? $O(n \log n)$

b) Ab welcher Datenmenge gilt die bessere Performance? $5n \log_{10} n = 25n$

$n = 10^5$

Folgende Aufgabe wird in der Übungsstunde bearbeitet:

$O(1)$

Aufgabe 4

Der offensichtliche Algorithmus zur Berechnung von x^n benötigt $n - 1$ Multiplikationen. Geben Sie einen schnelleren **rekursiven** Algorithmus für den Spezialfall an, dass der Exponent eine Zweierpotenz ist, d.h. $n = 2^m$, und berechnen Sie dessen Komplexität in O-Notation (gezählt wird hier die Anzahl der benötigten Multiplikationen).

$n = 2^m$

$x^n = x^{\frac{n}{2}} \cdot x^{\frac{n}{2}}$

$x^{2^{m-1}} = x^{2^m \cdot 2^{-1}} = x^{2^m \cdot \frac{1}{2}} = x^{\frac{n}{2}}$

$O(\log_2 n)$