

186.866 Algorithmen und Datenstrukturen VU**Übungsblatt 1**

PDF erstellt am: 11. März 2024

Deadline für dieses Übungsblatt ist **Montag, 18.3.2024, 20:00 Uhr**. Damit Sie für diese Übung Aufgaben anerkannt bekommen können, gehen Sie folgendermaßen vor:

1. Öffnen Sie den TUWEL-Kurs der Lehrveranstaltung *186.866 Algorithmen und Datenstrukturen (VU 5.5)* und navigieren Sie zum Abschnitt *Übungsblätter*.
2. Teilen Sie uns mit, welche Aufgaben Sie gelöst haben **und** welche gelösten Aufgaben Sie gegebenenfalls in der Übungseinheit präsentieren können. Gehen Sie dabei folgendermaßen vor:
 - Laden Sie Ihre Lösungen in einem einzigen PDF-Dokument in TUWEL hoch.
Link *Hochladen Lösungen Übungsblatt 1*
Button *Abgabe hinzufügen* bzw. *Abgabe bearbeiten*
PDF-Datei mit Lösungen hochladen und *Änderungen sichern*.
 - Kreuzen Sie an, welche Aufgaben Sie gegebenenfalls in der Übung präsentieren können. Die Lösungen der angekreuzten Aufgaben müssen im hochgeladenen PDF enthalten sein.
Link *Ankreuzen Übungsblatt 1*
Aufgaben entsprechend anhaken und *Änderungen speichern*.

Bitte beachten Sie:

- Bis zur Deadline können Sie sowohl Ihr hochgeladenes PDF, als auch Ihre angekreuzten Aufgaben beliebig oft überschreiben. Sollte kurz vor der Deadline etwas schief gehen (Ausfall TUWEL, Internet, Scanner, etc.) und Sie die Endversion mit allen gelösten Aufgaben nicht mehr hochladen können, haben Sie zumindest Ihre Lösungen teilweise schon hochgeladen und angekreuzt. Nach der Deadline ist keine Veränderung mehr möglich. Die Deadline ist strikt – es werden ausnahmslos keine Nachabgabeversuche (z.B. per E-Mail) akzeptiert.
- Sie können Ihre Lösungen entweder direkt in einem Textverarbeitungsprogramm erstellen und hochladen, oder aber auch gut leserliche Scans bzw. Fotos von handschriftlichen Ausarbeitungen hochladen (beachten Sie die maximale Dateigröße).
- Beachten Sie die Richtlinien für das An- und Aberkennen von Aufgaben. Details dazu finden Sie in den Folien der Vorbesprechung.

Aufgabe 1. Ordnen Sie folgende Funktionen nach Dominanz, beginnend mit der asymptotisch am schwächsten wachsenden. Es genügt die Funktionen zu reihen und die Reihung erklären zu können, ein Beweis der Gültigkeit der Relationen ist nicht erforderlich.

$$\log_2(n^{10}) \quad \frac{(n+1)!}{100} \quad 0.78^{2n} \quad \frac{n^{4.5}}{n^2} \quad \log_2(2^n) \quad 3^n \quad 2^n n \quad 30n^2 \quad 100 \cdot n! \quad 4^{n/2}$$

Aufgabe 2. Gegeben sind die folgende Funktionen:

$$f(n) = 30n^2(3^n + 2^n + 1^n)$$

$$g_1(n) = 3^n + n^2$$

$$g_2(n) = \begin{cases} 4^n & \text{falls } n \text{ durch } 7 \text{ teilbar ist} \\ 2^n & \text{sonst} \end{cases}$$

$$g_3(n) = \begin{cases} 4^n + 300n^2 + 400n & \text{falls } n < 10^{30} \\ 20 \cdot 3^n \cdot n^2 + 20 \cdot 3^n \cdot n + n^2 & \text{sonst} \end{cases}$$

Kreuzen Sie in der folgenden Tabelle die zutreffenden Felder an und begründen Sie Ihre Antworten:

$f(n)$ ist in	$\Theta(\cdot)$	$O(\cdot)$	$\Omega(\cdot)$	keines
$g_1(n)$				
$g_2(n)$				
$g_3(n)$				

Hinweis: Setzen Sie statt dem Punkt die entsprechend Funktion (g_1 , g_2 bzw. g_3) ein. Beispielsweise ist die Zelle links oben als „ $f(n)$ ist in $\Theta(g_1(n))$ “ zu lesen.

Aufgabe 3. Beweisen oder widerlegen Sie die folgenden Aussagen:

(a) $\max(f(n), g(n)) = \Theta(f(n) + g(n))$.

(b) $\min(f(n), g(n)) = \Theta(f(n) + g(n))$.

(c) Wenn $f_1(n) = O(g_1(n))$ und $f_2(n) = O(g_2(n))$, dann $f_1(n) \cdot f_2(n) = O(g_1(n) \cdot g_2(n))$.

Anmerkung: Wir nehmen an, dass alle Funktionen die nicht-negativen ganzen Zahlen als Definitions- und Wertebereich haben.

Aufgabe 4. Bestimmen Sie die Worst-Case und Best-Case Laufzeiten der unten angegebenen Algorithmen in Abhängigkeit von n in Θ -Notation.

```
(a)  s ← 0
      for i = 0, ..., n - 1
        A[i] = 2 · A[i]
        if ⌊A[i]/3⌋ < 0 then
          s ← s + A[i]
      return s
```

```
(b)  m ← 100
      for i = 0, ..., n - 1
        if A[i] = m then
          return i
      return -1
```

```
(c)  s ← 0
      i ← 0
      while i ≤ n - 1
        i ← i + 2
        for j = i, ..., n - 1
          s ← s + j
          for k = 0, ..., i - 1
            s ← s + k
      return s
```

```
(d)  a ← n
      for i = 0 ... n - 1
        d ← n
        while d > 0
          d ← ⌊d/2⌋
          a ← ⌈d/3⌉
      return a
```

Aufgabe 5. Bestimmen Sie die Laufzeiten **und** die Werte der Variablen a und b nach der Ausführung der unten angegebenen Algorithmen in Abhängigkeit von n in Θ -Notation. Verwenden Sie hierfür möglichst einfache Terme.

(a)

```

 $a \leftarrow 1$ 
 $b \leftarrow 1$ 
for  $c \leftarrow 1, \dots, \lfloor \log_4 n \rfloor$ 
     $a \leftarrow a + a$ 
     $b \leftarrow b + 1$ 
 $i \leftarrow 4^{4b}$ 
while  $i > 1$ 
     $i \leftarrow \lfloor \frac{i}{2} \rfloor$ 

```

(b)

```

 $a \leftarrow 1$ 
for  $i = 0, \dots, n - 1$ 
     $a \leftarrow a + a$ 
    for  $j = i, \dots, n - 1$ 
        if  $(j \bmod 2) = 0$  then
             $a \leftarrow a + j$ 
        for  $j = i + 1, \dots, n$ 
            if  $(j \bmod 2) = 1$  then
                 $a \leftarrow a - (j - 1)$ 
 $p \leftarrow \sqrt{a}$ 
 $b \leftarrow 0$ 
while  $p > 1$ 
     $p \leftarrow \lfloor p/2 \rfloor$ 
     $b \leftarrow b + p$ 

```
