Prof. Dr. Ir. Joost-Pieter Katoen
Christian Dehnert, Friedrich Gretz, Benjamin Kaminski, Thomas Ströder

Tutoraufgabe 1 (Rekursionsgleichungen):

Geben Sie die Rekursionsgleichung $T(d, l)$ für die Laufzeit des folgenden Algorithmus an. Gehen Sie dabei davon aus, dass die elementaren Operationen $\{+, -, \cdot, /, \sqrt{}\} \in \Theta(1)$ liegen.

```
float sierpinski(int depth, float length) {
  if (depth > 0) {
    return  sierpinski(depth-1, length/2) +
            sierpinski(depth-1, length/2) +
            sierpinski(depth-1, length/2);
  } else {
    // berechne die Flaeche des Dreiecks
    return √(length² − (length/2)²) * (length/2);
  }
}
```

Hinweis: `http://de.wikipedia.org/wiki/Sierpinski-Dreieck`

Lösung:

Die Rekursionsgleichung lautet

$$T(d, l) = \begin{cases} 3 \cdot T(d-1, \frac{l}{2}) + (6+2) & \text{falls } d > 0 \\ 7 & \text{sonst} \end{cases}$$

Der Parameter l hat also keinen Einfluss auf die Laufzeit.

Tutoraufgabe 2 (Substitutionsmethode):

Gegeben sei die folgende Rekursionsgleichung:

$$T(n) \quad = \quad \begin{cases} 4T(\frac{n}{2}) + \frac{n^2}{\log_2(n)}, & \text{falls } n > 1 \\ 1, & \text{sonst} \end{cases}$$

- Schätzen Sie mit Hilfe des Rekursionsbaumes eine (möglichst asymptotisch scharfe) obere Schranke für die Komplexitätsklasse der Laufzeit $T(n)$, d.h., geben Sie eine nicht-rekursive Funktion $f(n)$ mit $T(n) \in \mathcal{O}(f(n))$ an.
- Beweisen Sie mit Hilfe der Substitutionsmethode, dass $T(n) \in \mathcal{O}(f(n))$.

Hinweis: Die n–te Partialsumme der harmonischen Reihe ist definiert durch $H_n := \sum_{i=1}^{n} \frac{1}{i}$ und hat folgende Eigenschaften:

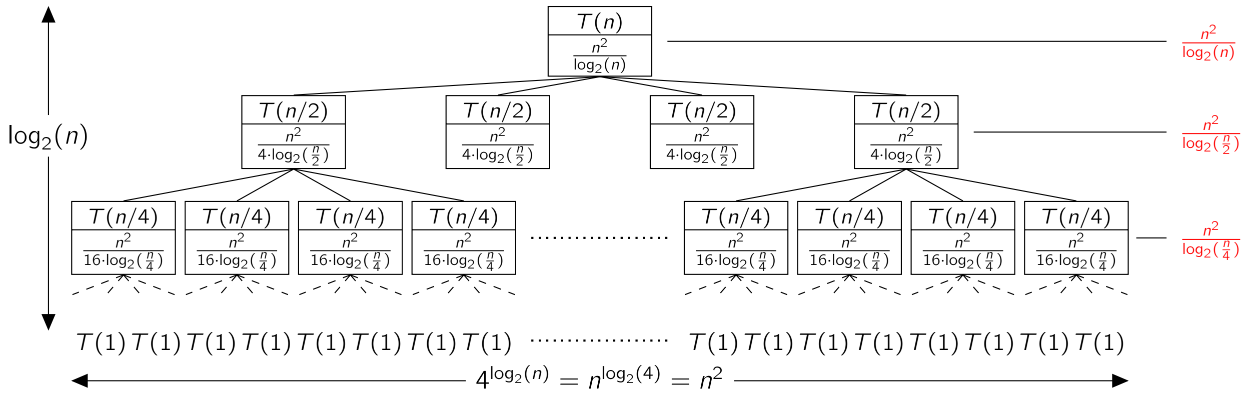
(i) $\forall n \geq 1.\ \log(n) < H_n$

(ii) $\forall n > 1.\ H_n - \log(n) \ < \ H_{n-1} - \log(n-1)$

Lösung:

Hinweis:
Das Mastertheorem ist nicht anwendbar, da $\frac{n^2}{\log_2(n)} \notin \mathcal{O}(n^{2-\epsilon})$, $\frac{n^2}{\log_2(n)} \notin \Theta(n^2)$ *und* $\frac{n^2}{\log_2(n)} \notin \Omega(n^{2+\epsilon})$.

Wir betrachten den Rekursionsbaum:



Aus dem Rekursionsbaum leiten wir folgende Abschätzung ab:

$$
\begin{aligned}
T(n) &= \left(\sum_{i=0}^{\log_2(n)-1} \frac{n^2}{\log_2(\frac{n}{2^i})}\right) + n^2 \\
&= n^2 \cdot \left(\sum_{i=0}^{\log_2(n)-1} \frac{1}{\log_2(n)-\log_2(2^i)}\right) + n^2 \\
&= n^2 \cdot \left(\sum_{i=0}^{\log_2(n)-1} \frac{1}{\log_2(n)-i}\right) + n^2 \\
&= n^2 \cdot \left(\sum_{i=1}^{\log_2(n)} \frac{1}{i}\right) + n^2 && \text{(Summe gedreht)} \\
&\approx n^2 \cdot \log(\log_2(n)) + n^2 && \text{(mit wachsendem } n\text{)}
\end{aligned}
$$

Daher stellen wir folgende Vermutung auf:

$$T(n) \;\in\; \mathcal{O}(n^2 \cdot \log(\log_2(n)))$$

Behauptung: $T(n) \in \mathcal{O}(n^2 \cdot \log(\log_2(n)))$

Beweis:
Wir müssen zeigen, dass:

$$\exists c_1 > 0.\ \exists n_0 \geq 0.\ \forall n \geq n_0.\ T(n) \leq c_1 \cdot n^2 \cdot \log(\log_2(n)))$$

Wähle $n_0 = 4$ und $c_1 = 10 \cdot \frac{1}{\log 2}$.

Induktionsanfang:

- $T(4) = 4 \cdot T(2) + \frac{4^2}{\log_2 4} = 4 \cdot \underbrace{(4 \cdot T(1) + \frac{2^2}{\log_2 2})}_{T(2)=8} + \frac{4^2}{\log_2 4} = 40$
  $\leq 10 \cdot \frac{1}{\log 2} \cdot 4^2 \cdot \log(\log_2(4)) = 10 \cdot 4^2 = 160$
- $T(5) = 4 \cdot T(3) + \frac{5^2}{\log_2 5} = 4 \cdot (4 \cdot T(2) + \frac{3^2}{\log_2 3}) + \frac{5^2}{\log_2 5} \approx 4 \cdot (4 \cdot 8 + 5.68) + 10.77 = 161.49$
  $\leq 10 \cdot \frac{1}{\log 2} \cdot 5^2 \cdot \log(\log_2(5)) \approx 303.83$
- $T(6) = 4 \cdot T(3) + \frac{6^2}{\log_2 6} = 4 \cdot (4 \cdot T(2) + \frac{3^2}{\log_2 3}) + \frac{6^2}{\log_2 6} \approx 4 \cdot (4 \cdot 8 + 5.68) + 13.93 = 164.65$
  $\leq 10 \cdot \frac{1}{\log 2} \cdot 6^2 \cdot \log(\log_2(6)) \approx 493.25$
- $T(7) = 4 \cdot T(4) + \frac{7^2}{\log_2 7} \approx 4 \cdot 40 + 17.45 = 177.45$
  $\leq 10 \cdot \frac{1}{\log 2} \cdot 7^2 \cdot \log(\log_2(7)) \approx 729.71$

Induktionsvoraussetzung: $\forall n_0 \leq m < n.\ T(m) \leq c_1 \cdot m^2 \cdot \log(\log_2(m))$
Induktionsschluss:

$$\begin{aligned}
T(n) &= 4 \cdot T(\tfrac{n}{2}) + \tfrac{n^2}{\log_2(n)} \\
&\leq 4 \cdot c_1 \cdot \tfrac{n^2}{4} \cdot \log(\log_2(\tfrac{n}{2})) + \tfrac{n^2}{\log_2(n)} && \Big|\ \text{Induktionsvoraussetzung} \\
&= c_1 \cdot n^2 \cdot \log(\log_2(n) - \log_2(2)) + \tfrac{n^2}{\log_2(n)} \\
&= c_1 \cdot n^2 \cdot \log\left(\log_2(n) - 1\right) + \tfrac{n^2}{\log_2(n)} \\
&< c_1 \cdot n^2 \cdot (\log\left(\log_2(n)\right) - \tfrac{1}{\log_2(n)}) + \tfrac{n^2}{\log_2(n)} && \Big|\ \text{(ii)} \Leftrightarrow \log(k-1) < H_{k-1} - H_k + \log(k) = -\tfrac{1}{k} + \log(k) \\
&= c_1 \cdot n^2 \cdot \log\left(\log_2(n)\right) - \tfrac{c_1 \cdot n^2}{\log_2(n)} + \tfrac{n^2}{\log_2(n)} \\
&= c_1 \cdot n^2 \cdot \log\left(\log_2(n)\right) + \underbrace{(1 - c_1) \cdot (\frac{n^2}{\log_2(n)})}_{<0 \text{ für } c_1=4 \text{ und } n \geq n_0 = 4} \\
&\leq c_1 \cdot n^2 \cdot \log\left(\log_2(n)\right)
\end{aligned}$$

□

Tutoraufgabe 3 (Variablentransformation):

Nutzen Sie die Methode der Variablentransformation um eine (asymptotisch scharfe) obere Schranke für die Rekursionsgleichung

$$T(n) = T(\sqrt[3]{n}) + 3$$

zu bestimmen.

Lösung:

a)

$$\begin{aligned}
& T(n) = T(\sqrt[3]{n}) + 3 && \mid \text{Variablentransformation } m = \log_2 n \\
\Rightarrow\ & T(2^m) = T(2^{\frac{m}{3}}) + 3 && \mid \text{Umbenennung } T(2^m) = S(m) \\
\Rightarrow\ & S(m) = S(\frac{m}{3}) + 3 && \mid \text{Lösung Rekursionsgleichung (z.B. MT): } S(m) = S(\frac{m}{3}) + 3 \\
\Rightarrow\ & S(m) \leq c \cdot \log_3 m && \mid m = \log_2 n \\
\Rightarrow\ & T(n) \leq c \cdot \log_3 \log_2 n \\
\Rightarrow\ & T(n) \in \mathcal{O}(\log \log_2 n))
\end{aligned}$$

Tutoraufgabe 4 (Master-Theorem):

Geben Sie für folgende Rekursionsgleichungen (mit Begründung) an, ob diese mit dem Master-Theorem gelöst werden können. Falls dies der Fall ist, geben Sie auch die resultierende Komplexitätsklasse an.

a) $T(n) = 4 \cdot T\left(\frac{n}{2}\right) + 5n$

b) $T(n) = 4 \cdot T\left(\frac{n}{4}\right) + 4n$

c) $T(n) = 2 \cdot T\left(\frac{n}{2}\right) + n^2$

d) $T(n) = 2 \cdot T(n)$

Lösung:

a) Es sind $b = 4$, $c = 2$ und $f(n) = 5n$. E wird nun wie folgt bestimmt:

$$E = \frac{\log(4)}{\log(2)} = \frac{2}{1} = 2$$

Damit gilt $n^E = n^2$. Wähle nun $\varepsilon = 1$. Damit gibt es nun zwei Konstanten $c' = 5$ und $n_0 = 1$, sodass gilt

$$\forall n \geq n_0 : f(n) = 5n \leq 5 \cdot n^{2-1} = c' \cdot n^{E-\varepsilon}$$

und damit gilt $f(n) \in \mathcal{O}(n^{E-\varepsilon})$. Somit findet der *erste Fall* des Master-Theorems Anwendung und es ergibt sich die Komplexitätsklasse $\Theta\left(n^E\right)$ für $T(n)$, also

$$T(n) \in \Theta(n^2)\ .$$

b) Es sind $b = 4$, $c = 4$ und $f(n) = 4n$. E wird nun wie folgt bestimmt:

$$E = \frac{\log(4)}{\log(4)} = \frac{2}{2} = 1$$

Damit gilt $n^E = n$. Damit gibt es nun drei Konstanten $c_1 = c_2 = 4$ und $n_0 = 1$, sodass gilt

$$\forall n \geq n_0 : c_1 \cdot n^E = 4 \cdot n^1 \leq \underbrace{4n}_{=f(n)} \leq 4 \cdot n^1 = c_2 \cdot n^E$$

und damit gilt $f(n) \in \Theta\left(n^E\right)$. Somit findet der *zweite Fall* des Master-Theorems Anwendung und es ergibt sich die Komplexitätsklasse $\Theta\left(n^E \cdot \log(n)\right)$ für $T(n)$, also

$$T(n) \in \Theta\big(n \cdot \log(n)\big)\ .$$

c) Es sind $b = 2$, $c = 2$ und $f(n) = n^2$. E wird nun wie folgt bestimmt:

$$E = \frac{\log(2)}{\log(2)} = \frac{1}{1} = 1$$

Damit gilt $n^E = n$. Wähle nun $\varepsilon = 1$. Damit gibt es nun zwei Konstanten $c' = 1$ und $n_0 = 1$, sodass gilt

$$\forall n \geq n_0 : f(n) = n^2 \geq 1 \cdot n^{1+1} = c' \cdot n^{E+\varepsilon}$$

und damit gilt $f(n) \in \Omega\left(n^{E+\varepsilon}\right)$. Somit fände der *dritte Fall* des Master-Theorems Anwendung. Es ist jedoch noch die zweite Bedingung $\exists\, 0 \leq d < 1\ \exists\, n_0\ \forall n \geq n_0 : b \cdot f\left(\frac{n}{c}\right) \leq d \cdot f(n)$ zu überprüfen. Wir wählen dafür $d = \frac{1}{2}$. Damit ergibt sich

$$b \cdot f\left(\frac{n}{c}\right) = 2 \cdot \frac{n^2}{2^2} = \frac{1}{2} \cdot n^2 \leq \frac{1}{2} \cdot n^2 = d \cdot f(n)\ ,$$

was offensichtlich für alle n gilt. Somit sind beide Bedingungen für den dritten Fall des Master-Theorems erfüllt und es ergibt sich die Komplexitätsklasse $\Theta\big(f(n)\big)$ für $T(n)$, also

$$T(n) \in \Theta(n^2)\ .$$

d) Es sind $b = 2$, $c = 1$ und $f(n) = 0$. E wird nun wie folgt bestimmt:

$$E = \frac{\log(2)}{\log(1)} = \frac{2}{0} = \textit{undefiniert}$$

Damit kann das Master-Theorem *nicht* angewendet werden.

Ein Algorithmus, dessen Laufzeit durch die vorliegende Rekursionsgleichung beschrieben werden kann, wäre beispielsweise der folgende:

```
int diverge(int k) {
    if (k > 0){
        diverge(k);
        diverge(k);
    }
}
```

Die Rekursionsgleichung $T(0) = 0$, $T(n) = 2 \cdot T(n)$ beschreibt die Anzahl der Aufrufe von `diverge`. Dieser Algorithmus terminiert jedoch für alle $k > 0$ nicht. Der Grund dafür ist, dass jeder Aufruf von `diverge` die Funktion `diverge` wieder mit demselben k aufgeruft. k wird also mit jedem Aufruf *nicht* verringert. Die „Komplexitätsklasse" des Algorithmus' ist sozusagen $\Theta(\infty)$.

Aufgabe 5 (Rekursionsgleichungen): (1+3+3+3+3+5* Punkte)

a) Im Folgenden betrachten wir das Problem der Türme von Hanoi. Gegeben sind 3 Stäbe. Initial sind die Stäbe 2 und 3 leer, und auf dem ersten Stab befinden sich n Scheiben paarweise unterschiedlicher Größe. Die Scheiben sind dabei in aufsteigender Größe angeordnet, die größte Scheibe liegt also unten und die kleinste oben (siehe Abbildung 1).

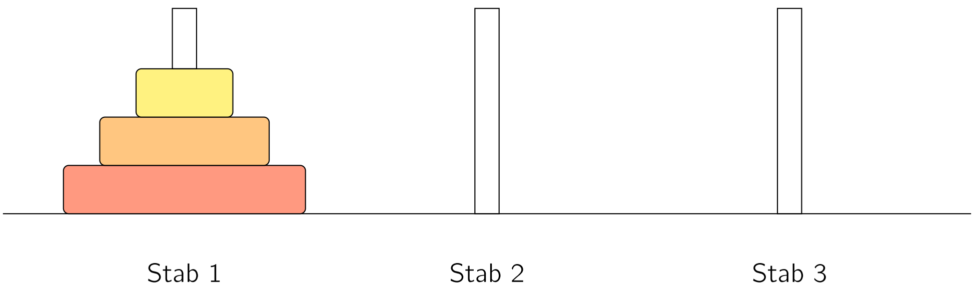


Abbildung 1: Die Türme von Hanoi mit $n = 3$ Scheiben.

Die Aufgabe ist nun, die Scheiben von Stab 1 auf Stab 3 zu verschieben. Dabei darf

- als elementare Operation immer nur die oberste Scheibe von einem Stab auf einen beliebigen anderen gelegt werden,
- niemals eine größere Scheibe auf einer kleineren liegen.

Ein Algorithmus, der eine das Problem löst, geht wie folgt vor:

```
// verschiebt die Scheiben von a nach c
// benutzt b als Puffer
verschiebe(int anzahlScheiben, Stab a, Stab b, Stab c) {
  if (anzahlScheiben > 0) {
    // verschiebe alle bis auf eine der Scheiben von a nach b
    verschiebe(anzahlScheiben - 1, a, c, b);
```

```
      // verschiebe oberste Scheibe von Stab a nach Stab c
      verschiebeObersteScheibe(a, c);

      // verschiebe die temporaer verschobenen Scheiben von b nach c
      verschiebe(anzahlScheiben - 1, b, a, c);
    }
  }
```

Geben Sie die Rekursionsgleichung für die Laufzeit dieses Algorithmus in Abhängigkeit der Anzahl Scheiben n an.

b) Zeigen Sie mit Hilfe der Substitutionsmethode, dass für

$$T(n) = \begin{cases} 2 \cdot T(\lceil \frac{n}{2} \rceil) + n & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

gilt $T(n) \in \Omega(n \cdot \log n)$.

Hinweis: Es darf hierbei nicht davon ausgegangen werden, dass die Rundung beim rekursiven Aufruf einfach vernachlässigt werden darf.

c) Zeigen Sie, dass die Lösung der Rekursionsgleichung

$$T(n) = \begin{cases} T(n-1) + \frac{1}{n} & \text{falls } n > 1 \\ 1 & \text{falls } n = 1 \end{cases}$$

in $\Theta(\log n)$ liegt.

d) Zeigen Sie mit Hilfe der Rekursionsbaum-Methode, dass die Lösung der Rekursionsgleichung

$$T(n) = \begin{cases} T(\frac{n}{3}) + T(\frac{2}{3}n) + cn & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

in $\Omega(n \log n)$ liegt für alle $c \in \mathbb{R}$, $c > 0$.

e) Raten Sie mit Hilfe der Rekursionsbaum-Methode die Komplexitätsklasse (Θ) der Lösung der Rekursionsgleichung

$$T(n) = \begin{cases} T(\frac{n}{2}) + 2T(\frac{n}{6}) + n & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

und beweisen Sie ihre Vermutung anschließend mit Hilfe der Substitutionsmethode.

Hinweis: Um eine Vermutung herzuleiten, kann es hilfreich sein, die geometrische Summe anzuwenden:

$$\sum_{i=0}^{n} a_0 q^i = a_0 \cdot \frac{1 - q^{n+1}}{1 - q} \text{ für } q \neq 1.$$

f) Diese Aufgabe ist eine **Bonusaufgabe**. Das bedeutet, dass die Punkte nicht zur Summe der erreichbaren Punkte beitragen, aber in dieser Aufgabe erhaltene Punkte ganz normal gutgeschrieben werden.

Zeigen Sie mit Hilfe der Substitutionsmethode, dass für

$$T(n) = \begin{cases} 2 \cdot T(\lfloor \frac{n}{2} \rfloor) + n & \text{falls } n > 1 \\ 1 & \text{sonst} \end{cases}$$

gilt $T(n) \in \Omega(n \cdot \log n)$.

Hinweis: Es darf hierbei nicht davon ausgegangen werden, dass die Rundung beim rekursiven Aufruf einfach vernachlässigt werden darf.

Lösung:

a) Die Rekursiongleichung ist gegeben durch $T(n) = 2 \cdot T(n-1) + 1$.

b) Zu zeigen ist jetzt, dass für geeignete c und n_0, $T(n) \geq cn \log_b n$ für alle $n \geq 2 =: n_0$ gilt. Wir wählen als Basis $b = 2$. Wir haben die Basisfälle:

$$T(2) = 2T(\left\lceil \frac{2}{2} \right\rceil) + 2 = 4 \geq 2c \log_2 2 \Leftrightarrow c \leq \underbrace{\frac{4}{2\log_2 2}}_{=2}$$

$$T(3) = 2T(\left\lceil \frac{3}{2} \right\rceil) + 2 = 2T(2) + 3 = 8 + 3 = 11 \geq 3c \log_2 3 \Leftrightarrow c \leq \underbrace{\frac{11}{3\log_2 3}}_{\approx 2.31}$$

Als Induktionsannahme erhalten wir für ein geeignetes c:

$$T(m) \geq cm \log_2(m) \text{ für alle } 2 \leq m < n$$

Somit ergibt sich für den Induktionsschritt:

$$\begin{aligned}
T(n) &= 2T(\left\lceil \frac{n}{2} \right\rceil) + n \\
&\overset{\text{(IV)}}{\geq} 2c \left\lceil \frac{n}{2} \right\rceil \log_2(\left\lceil \frac{n}{2} \right\rceil) + n \\
&\geq 2c\frac{n}{2} \log_2(\frac{n}{2}) + n \\
&= cn(\log_2(n) - \log_2(2)) + n \\
&= cn \log_2(n) - cn \log_2(2) + n \\
&= cn \log_2(n) - cn + n \qquad | \text{ für } 0 < c \leq 1 \\
&\geq cn \log_2(n)
\end{aligned}$$

Die Wahl $c = 1$ erfüllt die Bedingung für den Induktionsanfang und den Induktionsschritt. Damit ist die Aussage gezeigt.

c) Umschreiben der Rekursionsgleichung ergibt:

$$\begin{aligned}
T(n) &= T(n-1) + \frac{1}{n} \\
&= T(n-2) + \frac{1}{n-1} + \frac{1}{n} \\
&\vdots \\
&= T(n-(n-1)) + \frac{1}{n-(n-2)} + \cdots + \frac{1}{n-1} + \frac{1}{n} \\
&= T(1) + \frac{1}{2} + \cdots + \frac{1}{n-1} + \frac{1}{n} \\
&= \sum_{i=1}^{n} \frac{1}{i} \\
&= H_n
\end{aligned}$$

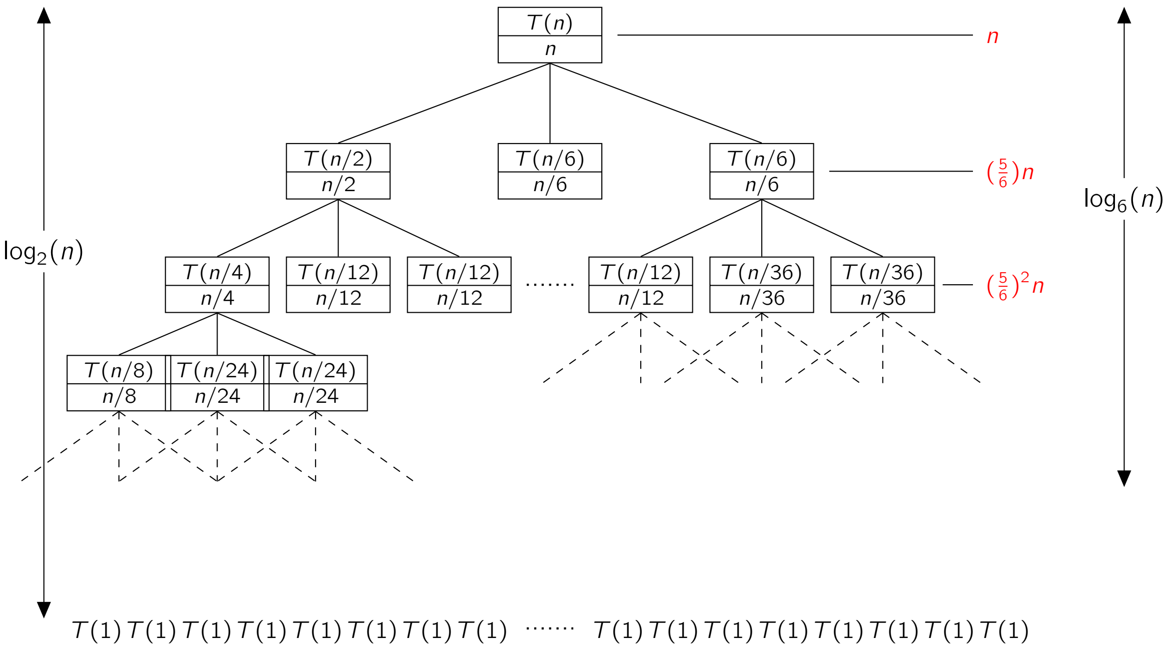
Aus den Hinweisen aus Tutoraufgabe 2 können wir ableiten, dass $\log(n) \leq H_n \leq 2\log(n)$ für $n > 2$. Das bedeutet unmittelbar, dass $H_n = T(n) \in \Theta(\log(n))$.

Leiten wir nun die Ungleichungen ab. $\log(n) \leq H_n$ für $n > 2$ ist durch Hinweis (i) gegeben. Es bleibt also zu zeigen, dass $H_n \leq 2\log(n)$ für $n > 2$:

$$\begin{aligned}
\underbrace{H_n - \log(n) \;<\; H_{n-1} - \log(n-1)}_{\text{(ii)}} \Leftrightarrow H_n &< \log(n) + \underbrace{H_{n-1} - \log(n-1)}_{\overset{\text{(ii)}}{\Rightarrow} < (H_{n-2} - \log(n-2))} \\
&< \log(n) + H_{n-2} - \log(n-2) \\
&< \dots \\
&< \log(n) + H_1 - \log(1) \\
&< \log(n) + \underbrace{1}_{\leq \log(n) \text{ für } n \geq 2} - 0 \\
&\leq 2\log(n)
\end{aligned}$$

d) Der resultierende Rekursionsbaum ist nicht vollständig (er besitzt Pfade unterschiedlicher Länge von der Wurzel zu einem Blatt). Für die Abschätzung einer unteren Schranke reicht es jedoch hier aus nur den "oberen Teil des Baums" zu betrachten. An der Rekursionsgleichung kann man ablesen, dass der kürzeste Weg von der Wurzel zu einem Blatt die Länge $\log_3 n$ hat. Da jede Ebene genau Kosten cn verursacht, folgt direkt, dass $T(n) \geq cn \log_3 n \in \Omega(n \log n)$.

e) Aus der Rekursionsgleichung konstruieren wir den folgenden Rekursionsbaum.



Als Hypothese erhalten wir

$$\sum_{i=0}^{\log n} \left(\frac{5}{6}\right)^i n = n \cdot \underbrace{\frac{1 - (\frac{5}{6})^{\log n + 1}}{1 - \frac{5}{6}}}_{\to 6 \text{ für } n \to \infty} \in \Theta(n)$$

Wir beweisen die Vermutung nun mit Hilfe der Substitutionsmethode. Als Induktionsbasis erhalten wir

$$T(1) = 1 \leq c$$

Nehmen wir daher nun an, dass $T(m) \leq cm$ für eine Konstante $c > 0$ und alle $1 \leq m < n$. Dann gilt

$$\begin{aligned} T(n) &= T(\frac{n}{2}) + 2T(\frac{n}{6}) + n \\ &\leq c\frac{n}{2} + 2c\frac{n}{6} + n \\ &= \frac{5}{6}cn + n \\ &= (1 + \frac{5}{6}c)n \qquad | \text{ für } c \geq 6 \\ &\leq cn \end{aligned}$$

Daher gilt $T(n) \in \mathcal{O}(n)$. Es bleibt zu zeigen, dass $T(n) \in \Omega(n)$:

$$T(n) = T(\frac{n}{2}) + 2T(\frac{n}{6}) + n \geq n$$

Schlussendlich folgt aus $T(n) \in \mathcal{O}(n)$ und $T(n) \in \Omega(n)$ die gewünschte Aussage $T(n) \in \Theta(n)$.

f) Zu zeigen ist jetzt, dass für geeignete $c > 0$ und n_0, $T(n) \geq cn \log_b n$ für alle $n \geq n_0$ gilt. Wir wählen als Basis $b = 2$. Wir haben die Basisfälle:

$$T(2) = 2T(\left\lfloor \frac{2}{2} \right\rfloor) + 2 = 4 \geq 2c\log_2 2 \Leftrightarrow c \leq \underbrace{\frac{4}{2\log_2 2}}_{=2}$$

$$T(3) = 2T(\left\lfloor \frac{3}{2} \right\rfloor) + 2 = 4 \geq 3c\log_2 3 \Leftrightarrow c \leq \underbrace{\frac{4}{3\log_2 3}}_{\approx 0.84}$$

Als Induktionsannahme erhalten wir für ein geeignetes c:

$$T(m) \geq cm\log_2(m) \text{ für alle } 2 \leq m < n$$

Somit ergibt sich für den Induktionsschritt:

$$\begin{aligned} T(n) &= 2T(\left\lfloor \frac{n}{2} \right\rfloor) + n \\ &\overset{(IV)}{\geq} 2c\left\lfloor \frac{n}{2} \right\rfloor \log_2(\left\lfloor \frac{n}{2} \right\rfloor) + n \\ &\geq 2c(\frac{n-1}{2})\log_2(\frac{n-1}{2}) + n \\ &= c(n-1)(\log_2(n) + \log_2(1 - \frac{1}{n}) - \log_2(2)) + n \\ &= cn\log_2(n) + \underbrace{cn\log_2(1 - \frac{1}{n}) - cn\log_2(2) - c\log_2(n) - c\log_2(1 - \frac{1}{n}) + c\log_2(2) + n}_{\overset{(*)}{\geq} 0 \text{ für } n \geq 2} \\ &\geq cn\log_2(n) \end{aligned}$$

Es bleibt zu zeigen, dass die Ungleichung (*) für ein geeignetes c erfüllt ist.

$$\begin{aligned}
&cn \underbrace{\log_2(1-\frac{1}{n})}_{\geq \log_2 \frac{1}{2} = -1 \text{ für } n \geq 2} - cn\log_2(2) - c\log_2(n) - c\log_2(1-\frac{1}{n}) + c\log_2(2) + n \\
&\geq -cn - cn - c\underbrace{\log_2(n)}_{\leq n} \underbrace{-c\underbrace{\log_2(1-\frac{1}{n})}_{\leq 0}}_{\geq 0} + c\log_2(2) + n \\
&= -cn - cn - cn + \underbrace{c}_{\geq 0} + n \\
&\geq -3cn + n \qquad \mid c < \frac{1}{3} \\
&\geq 0
\end{aligned}$$

Der Induktionsschritt und die Basisfälle können daher für $c = \frac{1}{4}$ durchgeführt werden. Damit ist die Aussage gezeigt.

Aufgabe 6 (Master-Theorem): (3+3+3+3 Punkte)

Geben Sie für folgende Rekursionsgleichungen (mit Begründung) an, ob diese mit dem Master-Theorem gelöst werden können. Falls dies der Fall ist, geben Sie auch die resultierende Komplexitätsklasse an.

a) $T(n) = 16 \cdot T\left(\frac{n}{4}\right) + 72n^{\sqrt{2}}$

b) $T(n) = 27 \cdot T\left(\frac{n}{9}\right) + n^2 + n^{\sqrt{2}}$

c) $T(n) = 16 \cdot T\left(\frac{n}{4}\right) + \frac{n^3+2n}{n}$

d) $T(n) = 4 \cdot T\left(\frac{n}{4}\right) + n \cdot \log_2(n)$

Lösung:

a) Es sind $b = 16$, $c = 4$ und $f(n) = 72n^{\sqrt{2}}$. E wird nun wie folgt bestimmt:

$$E \;=\; \frac{\log(16)}{\log(4)} \;=\; \frac{4}{2} \;=\; 2$$

Damit gilt $n^E = n^2$. Wähle nun $\varepsilon = 2 - \sqrt{2} \approx 0.586$. Damit gibt es nun zwei Konstanten $c' = 72$ und $n_0 = 1$, sodass gilt

$$\forall n \geq n_0 \colon f(n) \;=\; 72n^{\sqrt{2}} \;\leq\; 72 \cdot n^{2-(2-\sqrt{2})} \;=\; c' \cdot n^{E-\varepsilon}$$

und damit gilt $f(n) \in \mathcal{O}(n^{E-\varepsilon})$. Somit findet der *erste Fall* des Master-Theorems Anwendung und es ergibt sich die Komplexitätsklasse $\Theta\left(n^E\right)$ für $T(n)$, also

$$T(n) \in \Theta(n^2)\ .$$

b) Es sind $b = 27$, $c = 9$ und $f(n) = n^2 + n^{\sqrt{2}}$. E wird nun wie folgt bestimmt:

$$E \;=\; \frac{\log(27)}{\log(9)} \;=\; \frac{3}{2} \;=\; 1.5$$

Damit gilt $n^E = n^{1.5}$. Wähle nun $\varepsilon = 0.5$. Damit gibt es nun zwei Konstanten $c' = 1$ und $n_0 = 1$, sodass gilt

$$\forall n \geq n_0 \colon f(n) \;=\; n^2 + n^{\sqrt{2}} \;\geq\; n^2 \;=\; 1 \cdot n^{1.5+0.5} \;=\; c' \cdot n^{E+\varepsilon}$$

und damit gilt $f(n) \in \Omega(n^{E+\varepsilon})$. Somit fände der *dritte Fall* des Master-Theorems Anwendung. Es ist jedoch für den dritten Fall noch eine zweite Bedingung, nämlich $\exists\, 0{\leq}d{<}1\ \exists\, n_0\ \forall n{\geq}n_0 \colon b \cdot f\left(\frac{n}{c}\right) \leq d \cdot f(n)$, zu überprüfen:

$$\begin{aligned} b \cdot f\left(\frac{n}{c}\right) \;&=\; \frac{27n^2}{9^2} + \frac{27n^{\sqrt{2}}}{9^{\sqrt{2}}} \;\leq\; d \cdot n^2 + d \cdot n^{\sqrt{2}} \;=\; d \cdot (n^2 + n^{\sqrt{2}}) \\ \iff \qquad \frac{1}{3(1+n^{\sqrt{2}-2})} &+ \frac{27}{9^{\sqrt{2}}(n^{2-\sqrt{2}}+1)} \;\leq\; d\ . \end{aligned}$$

Die linke Seite der Ungleichung ist monoton fallend in n und ergibt für $n = 1$ ca. 0.77. Daher ist $d = 0.8$ eine sichere Wahl für d. Insgesamt sind also beide Bedingungen für den dritten Fall des Master-Theorems erfüllt und es ergibt sich die Komplexitätsklasse $\Theta\big(f(n)\big)$ für $T(n)$, also

$$T(n) \in \Theta\left(n^2 + n^{\sqrt{2}}\right) = \Theta(n^2)\ .$$

c) Es sind $b = 16$, $c = 4$ und $f(n) = \frac{n^3+2n}{n} = n^2 + 2$. E wird nun wie folgt bestimmt:

$$E \;=\; \frac{\log(16)}{\log(4)} \;=\; \frac{4}{2} \;=\; 2$$

Damit gilt $n^E = n^2$. Nun bestimmen wir den folgenden Grenzwert:

$$\lim_{n\to\infty} \frac{f(n)}{n^E} \;=\; \lim_{n\to\infty} \frac{n^2+2}{n^2} \;=\; \lim_{n\to\infty} 1 + \frac{2}{n^2} \;=\; 1$$

Da der Grenzwert konstant und echt größer 0 ist, gilt $f(n) \in \Theta\left(n^E\right)$. Somit findet der *zweite Fall* des Master-Theorems Anwendung und es ergibt sich die Komplexitätsklasse $\Theta\left(n^E \cdot \log(n)\right)$ für $T(n)$, also

$$T(n) \in \Theta\big(n^2 \cdot \log(n)\big)\ .$$

d) Es sind $b = 4$, $c = 4$ und $f(n) = n \cdot \log_2(n)$. E wird nun wie folgt bestimmt:

$$E \;=\; \frac{\log(4)}{\log(4)} \;=\; \frac{2}{2} \;=\; 1$$

Damit gilt $n^E = n$. Da $n \cdot \log_2(n)$ überproportional (d.h. um mehr als einen konstanten Faktor) schneller wächst als n, könnte – wenn überhaupt – nur der *dritte* Fall des Master-Theorems Anwendung finden. Wir bestimmen nun also für ein beliebiges $\varepsilon > 0$ den folgenden Grenzwert:

$$\lim_{n\to\infty} \frac{f(n)}{n^{E+\varepsilon}} \;=\; \lim_{n\to\infty} \frac{n \cdot \log_2(n)}{n^{1+\varepsilon}} \;=\; \lim_{n\to\infty} \frac{\ln(n)}{\ln(2) \cdot n^{\varepsilon}} \;\overset{\text{l'Hospital}}{=}\; \lim_{n\to\infty} \frac{1}{n \cdot \ln(2) \cdot n^{\varepsilon-1}} \;=\; \lim_{n\to\infty} \frac{1}{\ln(2) \cdot n^{\varepsilon}} \;=\; 0$$

Da der Grenzwert gleich 0 ist, gilt $f(n) \in o\left(n^{E+\varepsilon}\right)$ und somit $f(n) \notin \Omega\left(n^{E+\varepsilon}\right)$.

Daher trifft der dritte Fall des Master-Theorems *nicht* zu und das Master-Theorem ist für diese Rekursionsgleichung nicht anwendbar.