Einzelprüfung „Theoretische Informatik / Algorithmen (vertieft)“

Einzelprüfungsnummer 66115 / 2011 / Frühjahr

Thema 1 / Aufgabe 1

(*4 Rekursionsgleichungen*)

Stichwörter: Master-Theorem

Bestimmen Sie mit Hilfe des Master-Theorems für die folgenden Rekursionsgleichungen möglichst scharfe asymptotische untere und obere Schranken, falls das Master-Theorem anwendbar ist! Geben Sie andernfalls eine kurze Begründung, warum das Master-Theorem nicht anwendbar ist!

> **Exkurs: Master-Theorem**
>
> $$T(n) = a \cdot T\left(\frac{n}{b}\right) + f(n)$$
>
> $a =$ Anzahl der rekursiven Aufrufe, Anzahl der Unterprobleme in der Rekursion ($a \geq 1$).
>
> $\frac{1}{b} =$ Teil des Originalproblems, welches wiederum durch alle Unterprobleme repräsentiert wird, Anteil an der Verkleinerung des Problems ($b > 1$).
>
> $f(n) =$ Kosten (Aufwand, Nebenkosten), die durch die Division des Problems und die Kombination der Teillösungen entstehen. Eine von $T(n)$ unabhängige und nicht negative Funktion.
>
> Dann gilt:
>
> **1. Fall:** $T(n) \in \Theta\left(n^{\log_b a}\right)$
>
> falls $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$ für $\varepsilon > 0$
>
> **2. Fall:** $T(n) \in \Theta\left(n^{\log_b a} \cdot \log n\right)$
>
> falls $f(n) \in \Theta\left(n^{\log_b a}\right)$
>
> **3. Fall:** $T(n) \in \Theta(f(n))$
>
> falls $f(n) \in \Omega\left(n^{\log_b a + \varepsilon}\right)$ für $\varepsilon > 0$ und ebenfalls für ein c mit $0 < c < 1$ und alle hinreichend großen n gilt: $a \cdot f(\frac{n}{b}) \leq c \cdot f(n)$

(a) $T(n) = 16 \cdot T\left(\frac{n}{2}\right) + 40n - 6$

Lösungsvorschlag

Allgemeine Rekursionsgleichung:

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

Anzahl der rekursiven Aufrufe (a):

16

Anteil Verkleinerung des Problems (b):

um $\frac{1}{2}$ also $b = 2$

Laufzeit der rekursiven Funktion ($f(n)$):

$40n - 6$

Ergibt folgende Rekursionsgleichung:

$T(n) = 16 \cdot T\left(\frac{n}{2}\right) + 40n - 6$

1. Fall: $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$:

für $\varepsilon = 14$:

$f(n) = 40n - 6 \in \mathcal{O}\left(n^{\log_2 16 - 14}\right) = \mathcal{O}\left(n^{\log_2 2}\right) = \mathcal{O}(n)$

2. Fall: $f(n) \in \Theta\left(n^{\log_b a}\right)$:

$f(n) = 40n - 6 \notin \Theta\left(n^{\log_2 16}\right) = \Theta\left(n^4\right)$

3. Fall: $f(n) \in \Omega\left(n^{\log_b a + \varepsilon}\right)$:

$f(n) = 40n - 6 \notin \Omega\left(n^{\log_2 16 + \varepsilon}\right)$

$\Rightarrow T(n) \in \Theta\left(n^4\right)$

Berechne die Rekursionsgleichung auf WolframAlpha: WolframAlpha

(b) $T(n) = 27 \cdot T\left(\frac{n}{3}\right) + 3n^2 \log n$

Lösungsvorschlag

Allgemeine Rekursionsgleichung:

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

Anzahl der rekursiven Aufrufe (a):

27

Anteil Verkleinerung des Problems (b):

um $\frac{1}{3}$ also $b = 3$

Laufzeit der rekursiven Funktion ($f(n)$):

$3n^2 \log n$

Ergibt folgende Rekursionsgleichung:

$T(n) = 27 \cdot T\left(\frac{n}{3}\right) + 3n^2 \log n$

1. Fall: $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$:

$f(n) = 3n^2 \log n = n \in \mathcal{O}\left(n^{\log_3 27 - 24}\right) = \mathcal{O}\left(n^{\log_3 3}\right) = \mathcal{O}(n)$

2. Fall: $f(n) \in \Theta\left(n^{\log_b a}\right)$:

$f(n) = 3n^2 \log n = n \notin \Theta\left(n^{\log_3 27}\right) = \Theta\left(n^3\right)$

3. Fall: $f(n) \in \Omega\left(n^{\log_b a + \varepsilon}\right)$:

$f(n) = 3n^2 \log n = n \notin \Omega\left(n^{\log_3 27 + \varepsilon}\right)$

$\Rightarrow T(n) \in \Theta\left(n^3\right)$

Berechne die Rekursionsgleichung auf WolframAlpha: WolframAlpha

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

Lösungsvorschlag

Allgemeine Rekursionsgleichung:

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

Anzahl der rekursiven Aufrufe (a):

4

Anteil Verkleinerung des Problems (b):

um $\frac{1}{2}$ also $b = 2$

Laufzeit der rekursiven Funktion ($f(n)$):

$3n^2 + \log n$

Ergibt folgende Rekursionsgleichung:

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

1. Fall: $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$:

$f(n) = 3n^2 + \log n = n^2 = n \notin \mathcal{O}\left(n^{\log_2 4 - \varepsilon}\right)$

2. Fall: $f(n) \in \Theta\left(n^{\log_b a}\right)$:

$f(n) = 3n^2 + \log n = n^2 = n \in \Theta\left(n^{\log_2 4}\right) = \Theta\left(n^2\right)$

3. Fall: $f(n) \in \Omega\left(n^{\log_b a + \varepsilon}\right)$:

$f(n) = 3n^2 + \log n = n^2 = n \notin \Omega\left(n^{\log_2 4 + \varepsilon}\right)$

$\Rightarrow T(n) \in \Theta\left(n^2 \log n\right)$

Berechne die Rekursionsgleichung auf WolframAlpha: WolframAlpha

(d) $T(n) = 4 \cdot T\left(\frac{n}{2}\right) + 100 \log n + \sqrt{2n} + n^{-2}$

Lösungsvorschlag

Allgemeine Rekursionsgleichung:

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

Anzahl der rekursiven Aufrufe (a):

4

Anteil Verkleinerung des Problems (b):

um $\frac{1}{2}$ also $b = 2$

Laufzeit der rekursiven Funktion ($f(n)$):

$100 \log n + \sqrt{2n} + n^{-2}$

Ergibt folgende Rekursionsgleichung:

$T(n) = 4 \cdot T\left(\frac{n}{2}\right) + 100 \log n + \sqrt{2n} + n^{-2}$

1. Fall: $f(n) \in \mathcal{O}\left(n^{\log_b a - \varepsilon}\right)$:

$f(n) = 100 \log n + \sqrt{2n} + n^{-2} = n \in \mathcal{O}\left(n^{\log_2 4 - 2}\right) = \mathcal{O}(n)$

2. Fall: $f(n) \in \Theta\big(n^{\log_b a}\big)$:

$$f(n) = 100 \log n + \sqrt{2n} + n^{-2} = n \notin \Theta\big(n^{\log_2 4}\big) = \Theta\big(n^2\big)$$

3. Fall: $f(n) \in \Omega\big(n^{\log_b a + \varepsilon}\big)$:

$$f(n) = 100 \log n + \sqrt{2n} + n^{-2} = n \notin \Omega\big(n^{\log_2 4 + \varepsilon}\big)$$

$\Rightarrow T(n) \in \Theta\big(n^2\big)$

Berechne die Rekursionsgleichung auf WolframAlpha: WolframAlpha

Die Bschlangaul-Sammlung
Hermine Bschlangaul and Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.