





Grundzüge der Informatik 1

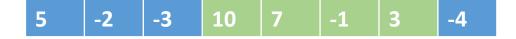
Vorlesung 7 - flipped classroom

Teile & Herrsche Verfahren

Aufgabe 1

- Entwickeln Sie einen Teile&Herrsche Algorithmus für das Teilsummenproblem
- Es genügt, wenn Sie den Wert der optimalen Lösung bestimmen
- Stellen Sie eine Laufzeitrekursion auf. Können Sie diese Auflösen?

Beispiel





Laufzeitanalyse - Wo ist der Fehler?

Aufgabe 2: Falsche Behauptung

Mergesort hat Laufzeit O(n).

Falscher Beweis

- Wir zeigen die Aussage nur für den Fall, dass n eine Zweierpotenz ist.
- Die Laufzeit für T(2) ist O(1). Wir zeigen per Induktion, T(n)=O(n) für alle n≥2.
- Induktionsanfang: für n=2 gilt T(2) = O(1).
- Induktionsannahme: Für Eingabelänge m<n, m Zweierpotenz, ist die Laufzeit T(m) = O(m).
- Induktionsschluss: Sei n eine Zweierpotenz. Es gilt T(n) = 2 T(n/2) + O(n).
 Nach Induktionsannahme gilt T(n) = 2 O(n) + O(n) = O(n)
- Also gilt T(n) = O(n).



- Wo ist der Fehler?

Aufgabe 2: Falsche Behauptung

Mergesort hat Laufzeit O(n).

Wo liegt der Fehler?

- A) Im Induktionsanfang
- B) In der Induktionsannahme
- C) Im Induktionsschluss
- D) Der Beweis ist korrekt. Die Behauptung stimmt nicht, wenn n keine Zweierpotenz ist.

Falscher Beweis

- Wir zeigen die Aussage nur für den Fall, dass n eine Zweierpotenz ist.
- Die Laufzeit für T(2) ist O(1). Wir zeigen per Induktion, T(n)=O(n) für alle n≥2.
- Induktionsanfang: für n=2 gilt T(2) = O(1).
- Induktionsannahme: Für Eingabelänge m<n, m Zweierpotenz, ist die Laufzeit T(m) = O(m).
- Induktionsschluss: Sei n eine Zweierpotenz. Es gilt T(n) = 2 T(n/2) + O(n). Nach Induktionsannahme gilt T(n) = 2 O(n) + O(n) = O(n)
- Also gilt T(n) = O(n).



Teile & Herrsche Algorithmen

Wodurch unterscheiden sich Teile & Herrsche Algorithmen?

- Die Anzahl der Teilprobleme
- Die Größe der Teilprobleme
- Den Algorithmus für das Zusammensetzen der Teilprobleme
- Den Rekursionsabbruch

Wann lohnt sich Teile & Herrsche?

Kann durch Laufzeitanalyse vorhergesagt werden



- Rekursionen

Laufzeiten der Form

$$T(n) = a T(n/b) + f(n)$$

$$(und T(1) = O(1))$$



- Rekursionen

Laufzeiten der Form

$$T(n) = aT(n/b) + f(n)$$

$$(und T(1) = O(1))$$



- Rekursionen

Laufzeiten der Form

$$T(n) = a T(n/b) + f(n)$$

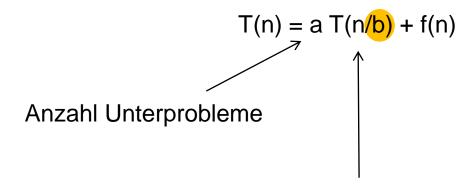
Anzahl Unterprobleme

$$(und T(1) = O(1))$$



Laufzeitanalyse - Rekursionen

Laufzeiten der Form

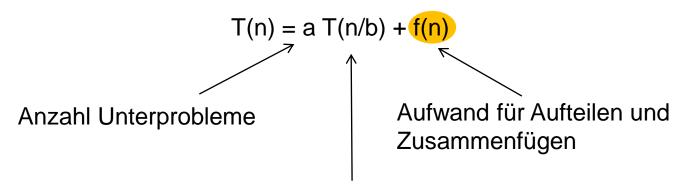


(und T(1) = O(1)) Größe der Unterprobleme (bestimmt Höhe des Rekursionsbaums)



Laufzeitanalyse - Rekursionen

Laufzeiten der Form



(und T(1) = O(1)) Größe der Unterprobleme (bestimmt Höhe des Rekursionsbaums)



- Rekursionen

Aufgabe 3

- Betrachten Sie folgende Laufzeitrekursion
- $T(n) = 2 T(n/2) + n^2$
- T(1) = 1
- Finden Sie eine Lösung für diese Rekursion
- Zeigen Sie die Korrektheit Ihrer Lösung. Sie können annehmen, dass neine Zweierpotenz ist.

