GBI Tutorium Nr. 2^5

Tutorium 8

Dominik Muth - dominik.muth@student.kit.edu | 12. Dezember 2012

INSTITUT FÜR INFORMATIK

Outline/Gliederung

Karlsruher Institut für Technologie

1. Wiederholung

2. Quantitative Aspekte von Algorithmen
   - Θ – *Notation*
   - O – *Notation*
   - Ω – *Notation*

3. Aufgaben

4. Fragen

Überblick

1. Wiederholung
2. Quantitative Aspekte von Algorithmen
3. Aufgaben
4. Fragen

Wiederholung - Quiz

- In Adjazenzmatrizen sind immer schneller als Adjazenzlisten, benötigen aber mehr Speicher.
- Hat ein Knoten x keine Ausgehenden Kanten, so steht in der Wegematrix in der x. Zeile nur 0en.
- Über Matrizenmultiplikation und addition lässt sich eine Wegematrix konstruieren.
- Die 2-Erreichbarkeitsrelation sagt uns nichts darüber aus, ob es einen Pfad der Länge 1 von einem Knoten zu einem andern gibt.

Wiederholung - Quiz

- In Adjazenzmatrizen sind immer schneller als Adjazenzlisten, benötigen aber mehr Speicher. *X*
- Hat ein Knoten x keine Ausgehenden Kanten, so steht in der Wegematrix in der x. Zeile nur 0en.
- Über Matrizenmultiplikation und addition lässt sich eine Wegematrix konstruieren.
- Die 2-Erreichbarkeitsrelation sagt uns nichts darüber aus, ob es einen Pfad der Länge 1 von einem Knoten zu einem andern gibt.

Wiederholung - Quiz

- In Adjazenzmatrizen sind immer schneller als Adjazenzlisten, benötigen aber mehr Speicher. *X*
- Hat ein Knoten x keine Ausgehenden Kanten, so steht in der Wegematrix in der x. Zeile nur 0en. *X*
- Über Matrizenmultiplikation und addition lässt sich eine Wegematrix konstruieren.
- Die 2-Erreichbarkeitsrelation sagt uns nichts darüber aus, ob es einen Pfad der Länge 1 von einem Knoten zu einem andern gibt.

Wiederholung - Quiz

- In Adjazenzmatrizen sind immer schneller als Adjazenzlisten, benötigen aber mehr Speicher. X
- Hat ein Knoten x keine Ausgehenden Kanten, so steht in der Wegematrix in der x. Zeile nur 0en. X
- Über Matrizenmultiplikation und addition lässt sich eine Wegematrix konstruieren. $\surd$
- Die 2-Erreichbarkeitsrelation sagt uns nichts darüber aus, ob es einen Pfad der Länge 1 von einem Knoten zu einem andern gibt. $\surd$

Wiederholung - Aufgaben

Karlsruher Institut für Technologie

A^2

Gegeben sei A mit: $A = \begin{array}{c c} & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$

- Zeichnen Sie den Graphen
- Berechnen Sie A^2

Wiederholung - Aufgaben

Karlsruher Institut für Technologie

A^2

Gegeben sei A mit: $A = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$

- Zeichnen Sie den Graphen
- Berechnen Sie A^2

Wiederholung - Fragen

Karlsruher Institut für Technologie

- Was ist eine Äquivalenzrelation?

Wiederholung - Fragen

- Was ist eine Äquivalenzrelation?
- Was ist eine symmetrische Relation?

Wiederholung - Fragen

- Was ist eine Äquivalenzrelation?
- Was ist eine symmetrische Relation?
- Was ist eine reflexive Relation?

Wiederholung - Fragen

- Was ist eine Äquivalenzrelation?
- Was ist eine symmetrische Relation?
- Was ist eine reflexive Relation?
- Was ist eine transitive Relation?

Überblick

1 Wiederholung

2 Quantitative Aspekte von Algorithmen
- Θ – *Notation*
- O – *Notation*
- Ω – *Notation*

3 Aufgaben

4 Fragen

Quantitative Aspekte von Algorithmen

Karlsruher Institut für Technologie

?

- Was verstehen wir unter quantitativen Aspekten?
- Wozu benötigen wir Laufzeitabschätzungen?
- Wie geben wir die Laufzeit an, in Sekunden?

Quantitative Aspekte von Algorithmen

Karlsruher Institut für Technologie

?

- Was verstehen wir unter quantitativen Aspekten?
- Wozu benötigen wir Laufzeitabschätzungen?
- Wie geben wir die Laufzeit an, in Sekunden?

Quantitative Aspekte von Algorithmen

?

- Was verstehen wir unter quantitativen Aspekten?
- Wozu benötigen wir Laufzeitabschätzungen?
- Wie geben wir die Laufzeit an, in Sekunden?

Θ – *Notation*

Definition

Mit dem Theta-Kalkül lässt sich dich Laufzeit eines Algorithmus darstellen.

$\Theta(f)$ ist eine Äquivalenzklasse mit der Äquivalenzrelation $\asymp$.

Einschub Äquivalenzklasse/Relation

$\asymp$

$\asymp$ steht für die Äquivalenzrelation des Asymptotisch gleich schnellen Wachstums.

$f \asymp g \Leftrightarrow \exists c, c' \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n > n' : cf(n) \leq g(n) \leq c'f(n)$

???

Annahme, wir haben eine Funktion $f = n^2$ und eine Funktion $g = n^2 + n$

Dann gilt $f \asymp g$ da es ein c und ein c' gibt, für welches gilt:

$\forall n > 1 : cf(n) \leq g(n) \leq c'f(n)$

ür welches c und c' wäre dies z.B. der Fall?

Einschub Äquivalenzklasse/Relation

Karlsruher Institut für Technologie

$\asymp$

$\asymp$ steht für die Äquivalenzrelation des Asymptotisch gleich schnellen Wachstums.

$f \asymp g \Leftrightarrow \exists c, c' \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n > n' : cf(n) \leq g(n) \leq c' f(n)$

???

Annahme, wir haben eine Funktion $f = n^2$ und eine Funktion $g = n^2 + n$

Dann gilt $f \asymp g$ da es ein c und ein c' gibt, für welches gilt:

$\forall n > 1 : cf(n) \leq g(n) \leq c' f(n)$

ür welches c und c' wäre dies z.B. der Fall?

Einschub Äquivalenzklasse/Relation

Karlsruher Institut für Technologie

$\asymp$

$\asymp$ steht für die Äquivalenzrelation des Asymptotisch gleich schnellen Wachstums.

$f \asymp g \Leftrightarrow \exists c, c' \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n > n' : cf(n) \leq g(n) \leq c' f(n)$

???

Annahme, wir haben eine Funktion $f = n^2$ und eine Funktion $g = n^2 + n$

Dann gilt $f \asymp g$ da es ein c und ein c' gibt, für welches gilt:

$\forall n > 1 : cf(n) \leq g(n) \leq c' f(n)$

ür welches c und c' wäre dies z.B. der Fall?

Einschub Äquivalenzklasse/Relation

Karlsruher Institut für Technologie

$\asymp$

$\asymp$ steht für die Äquivalenzrelation des Asymptotisch gleich schnellen Wachstums.

$f \asymp g \Leftrightarrow \exists c, c' \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n > n' : cf(n) \leq g(n) \leq c' f(n)$

???

Annahme, wir haben eine Funktion $f = n^2$ und eine Funktion $g = n^2 + n$

Dann gilt $f \asymp g$ da es ein c und ein c' gibt, für welches gilt:

$\forall n > 1 : cf(n) \leq g(n) \leq c' f(n)$

Für welches c und c' wäre dies z.B. der Fall?

Einschub Äquivalenzklasse/Relation

Die Äquivalenzklasse Θ

$\Theta(f)$ ist die Menge aller Funktionen g, welche asymptotisch genauso schnell wachsen wie f, also:

$$\Theta(f) = \{g \mid f \asymp g\}$$

Θ – *Notation*

Wie bestimmen wir, ob eine Funktion asymptotisch so schnell wächst, wie eine andere?

Θ – *Notation*

Wie bestimmen wir, ob eine Funktion asymptotisch so schnell wächst, wie eine andere?

Über den Grad von Polynomen (höchster Exponent)

Θ – *Notation*

Wie bestimmen wir, ob eine Funktion asymptotisch so schnell wächst, wie eine andere?

Über den Grad von Polynomen (höchster Exponent)

Warum funktioniert das so einfach?

Θ – *Notation*

Wie bestimmen wir, ob eine Funktion asymptotisch so schnell wächst, wie eine andere?

Über den Grad von Polynomen (höchster Exponent)

Warum funktioniert das so einfach?

Gegenbeispiel für verschiedene Grade:
$n^x \not\asymp n^{x+1}$ da es kein c gibt, für welches n^x größer ist als n^{x+1} für ein beliebig großes n.

Aufgabe

Zeigen sie: $log_2(n) \in \Theta(log_8(n))$

$O - Notation$

Karlsruher Institut für Technologie

Definition

In $O(f)$ liegen alle Funktionen, welche asymptotisch maximal so schnell Wachsen wie f.
$g \in O(f) \Rightarrow \exists c \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n : cf(n) \geq g$

Denken!!!

Was liegt in $O(1)$?

O – *Notation*

Karlsruher Institut für Technologie

Definition

In $O(f)$ liegen alle Funktionen, welche asymptotisch maximal so schnell Wachsen wie f.

$g \in O(f) \Rightarrow \exists c \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n : cf(n) \geq g$

Denken!!!

Was liegt in $O(1)$?

Aufgabe

Es sei *a* ein Array der Länge n.
Gegeben sei folgender Algorithmus:

$x \leftarrow 0$
for $i \leftarrow 0$ **to** $n-1$ **do**
  for $j \leftarrow 0$ **to** $n-1$ **do**
    $x \leftarrow x + a[j]$
  od
  for $k \leftarrow 1$ **to** n^2 **do**
    $x \leftarrow x + k \cdot a[i]$
  od
od

Schätzen Sie die Laufzeit möglichst passend im O-Kalkül ab.

Ω – *Notation*

Karlsruher Institut für Technologie

Definition

In $\Omega(f)$ liegen alle Funktionen, welche asymptotisch mindestens so schnell Wachsen wie f.

$g \in O(f) \Rightarrow \exists c \in \mathbb{R}^+ : \exists n' \in \mathbb{N}_0 \forall n : cf(n) \leq g$

Denken!!!

Was liegt in $\Omega(1)$?

Ω – *Notation*

Definition

In $\Omega(f)$ liegen alle Funktionen, welche asymptotisch mindestens so schnell Wachsen wie f.

$g \in O(f) \Rightarrow \exists c \in \mathbb{R}^{+} : \exists n' \in \mathbb{N}_0 \forall n : cf(n) \leq g$

Denken!!!

Was liegt in $\Omega(1)$?

Hinweise

Achtung

Oftmals liest man in Büchern:

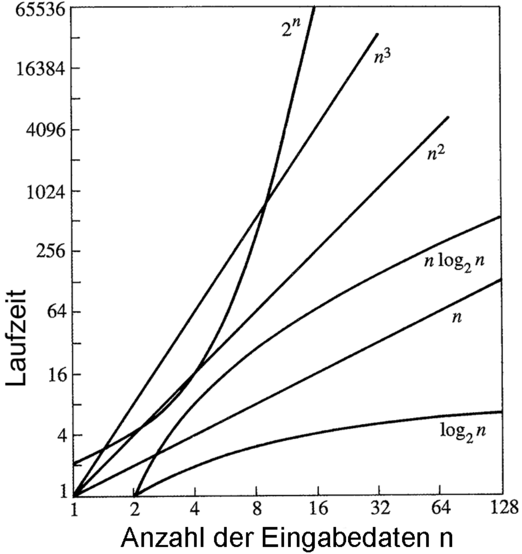
$g(n) = O(f(n))$

Das ist falsch, da das linke eine Funktion und das rechte eine Menge ist!

Ich zieh dafür Punkte ab!

Richtig ist: $g(n) \in O(f(n))$

Hinweise



65536
16384
4096
1024
256
64
16
4
1
Laufzeit
2^n
n^3
n^2
$n \log_2 n$
n
$\log_2 n$
1
2
4
8
16
32
64
128
Anzahl der Eingabedaten n

Überblick

1. Wiederholung
2. Quantitative Aspekte von Algorithmen
3. Aufgaben
4. Fragen

Aufgaben

Bestimmen Sie die Laufzeit des Warshall-Algorithmus:

```
for i ← 0 to n − 1 do
    for j ← 0 to n − 1 do
        if i = j then
            W[i, j] ← 1;
        else
            W[i, j] ← A[i, j]
        end if
    od
od
for k ← 0 to n − 1 do
    for i ← 0 to n − 1 do
        for j ← 0 to n − 1 do
            W[i, j] ← max(W[i, j]; min(W[i, k]; W[k, j]))
        od
    od
od
```

Aufgaben

Zeigen oder widerlegen Sie:

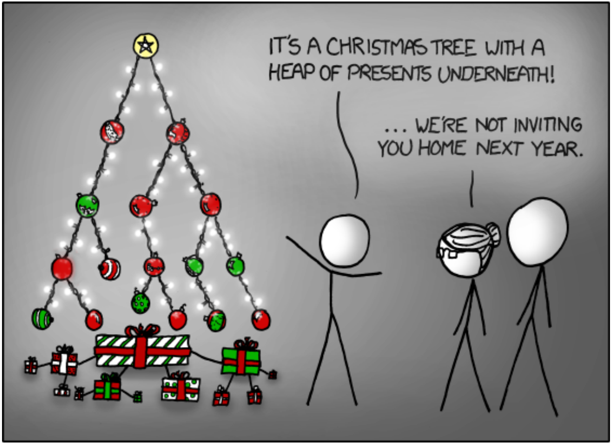
- $\frac{n^3+2n}{2n+1} \in 0(n^2)$
- $n! \in \Omega(n^2)$

Überblick

Fragen

- Fragen zum Stoff?
- Fragen zum nächsten Übungsblatt?
- Generelle Fragen?
- Feedback?

Frohe Weihnachten



source : http : / / imgs.xkcd.com / comics / tree.png